



**University of
Zurich**^{UZH}

**Zurich Open Repository and
Archive**

University of Zurich
University Library
Strickhofstrasse 39
CH-8057 Zurich
www.zora.uzh.ch

Year: 2015

Joint 3-D vessel segmentation and centerline extraction using oblique Hough forests with steerable filters

Schneider, Matthias ; Hirsch, Sven ; Weber, Bruno ; Székely, Gábor ; Menze, Bjoern H

Abstract: Contributions We propose a novel framework for joint 3-D vessel segmentation and centerline extraction. The approach is based on multivariate Hough voting and oblique random forests (RFs) that we learn from noisy annotations. It relies on steerable filters for the efficient computation of local image features at different scales and orientations. Experiments We validate both the segmentation performance and the centerline accuracy of our approach both on synthetic vascular data and four 3-D imaging datasets of the rat visual cortex at 700 nm resolution. First, we evaluate the most important structural components of our approach: (1) Orthogonal subspace filtering in comparison to steerable filters that show, qualitatively, similarities to the eigenspace filters learned from local image patches. (2) Standard RF against oblique RF. Second, we compare the overall approach to different state-of-the-art methods for (1) vessel segmentation based on optimally oriented flux (OOF) and the eigenstructure of the Hessian, and (2) centerline extraction based on homotopic skeletonization and geodesic path tracing. Results Our experiments reveal the benefit of steerable over eigenspace filters as well as the advantage of oblique split directions over univariate orthogonal splits. We further show that the learning-based approach outperforms different state-of-the-art methods and proves highly accurate and robust with regard to both vessel segmentation and centerline extraction in spite of the high level of label noise in the training data.

DOI: <https://doi.org/10.1016/j.media.2014.09.007>

Posted at the Zurich Open Repository and Archive, University of Zurich

ZORA URL: <https://doi.org/10.5167/uzh-120359>

Journal Article

Accepted Version



The following work is licensed under a Creative Commons: Attribution-NonCommercial-NoDerivatives 4.0 International (CC BY-NC-ND 4.0) License.

Originally published at:

Schneider, Matthias; Hirsch, Sven; Weber, Bruno; Székely, Gábor; Menze, Bjoern H (2015). Joint 3-D vessel segmentation and centerline extraction using oblique Hough forests with steerable filters. *Medical Image Analysis*, 19(1):220-249.

DOI: <https://doi.org/10.1016/j.media.2014.09.007>

Joint 3-D Vessel Segmentation and Centerline Extraction Using Oblique Hough Forests with Steerable Filters

Matthias Schneider^{a,b,*}, Sven Hirsch^{a,c}, Bruno Weber^b, Gábor Székely^a,
Bjoern H. Menze^{a,d}

^aComputer Vision Laboratory, ETH Zurich, Sternwartstrasse 7, 8092 Zurich, Switzerland

^bInstitute of Pharmacology and Toxicology, University of Zurich, Winterthurerstrasse 190,
8057 Zurich, Switzerland

^cInstitute of Applied Simulation, ZHAW Wädenswil, 8820 Wädenswil, Switzerland

^dInstitute for Advanced Study and Department of Computer Science, TU Munich,
Lichtenbergstrasse 2a, 85748 Garching, Germany

Abstract

Contributions. We propose a novel framework for joint 3-D vessel segmentation and centerline extraction. The approach is based on multivariate Hough voting and oblique random forests (RFs) that we learn from noisy annotations. It relies on steerable filters for the efficient computation of local image features at different scales and orientations.

Experiments. We validate both the segmentation performance and the centerline accuracy of our approach both on synthetic vascular data and four 3-D imaging datasets of the rat visual cortex at 700 nm resolution. First, we evaluate the most important structural components of our approach: (1) Orthogonal subspace filtering in comparison to steerable filters that show, qualitatively, similarities to the eigenspace filters learned from local image patches. (2) Standard RF against oblique RF. Second, we compare the overall approach to different state-of-the-art methods for (1) vessel segmentation based on optimally oriented flux (OOF) and the eigenstructure of the Hessian, and (2) centerline extraction based on homotopic skeletonization and geodesic path tracing.

Results. Our experiments reveal the benefit of steerable over eigenspace filters as well as the advantage of oblique split directions over univariate orthogonal splits. We further show that the learning-based approach outperforms different state-of-the-art methods and proves highly accurate and robust with regard to

*Corresponding author.

Address: ETF D114.1, Sternwartstrasse 7, 8092 Zurich, Switzerland

Email: schneider@vision.ee.ethz.ch

Phone: +41 44 63 20379

Fax: +41 44 63 21199

Web: <http://www.vision.ee.ethz.ch/~schneima>

both vessel segmentation and centerline extraction in spite of the high level of label noise in the training data.

Keywords: vessel segmentation, centerline extraction, steerable filters, oblique random forest, multivariate Hough voting

1. Introduction

Segmentation and analysis of tubular structures such as blood vessels, in particular, play a crucial role for numerous medically oriented applications and have attracted a lot of attention in the field of medical image processing. The multi-scale nature of vessels, image noise and contrast inhomogeneities make it a challenging task. In this context, a large variety of methods have been developed exploiting photometric and structural properties of tubular structures.

1.1. Related Work

Vessel Segmentation. Vessel segmentation is an established field in biomedical image processing, see for example Kirbas and Quek (2004) and Lesage et al. (2009) providing extensive reviews. Many of them are notably tailored to specific applications and imaging modalities. Rather simple methods for vessel detection, e.g., absolute or locally adaptive thresholding (Otsu, 1979; Canny, 1983), are regularly used in practice due to their conceptual simplicity and computational efficiency but they are a serious source of error and require careful parameter selection. More sophisticated segmentation techniques can roughly be divided into two groups. One group pursues a top-down strategy by iteratively propagating segmentation labels starting at set of seeds towards distal branches by means of, e.g., region growing (Martínez-Pérez et al., 1999; Lo et al., 2010), active contours (Lorigo et al., 2001), particle filtering (Lesage et al., 2008; Florin et al., 2006), or path tracing (Zhou et al., 2007; Schneider and Sundar, 2010). The design and choice of an appropriate energy or posterior density term to guide the evolution of the segmentation is crucial and usually involves strong assumptions on the underlying structures to be detected. Similarly, elaborate break criteria are required to prevent the segmentation from leaking into the background, particularly for data with a low signal to noise ratio. Another group of methods follows the bottom-up paradigm aiming at optimizing a global neighborhood graph in order to incorporate spatial context (Türetken et al., 2012; Rempfler et al., 2014). The graph is commonly defined on the voxel locations based on a likelihood for a voxel belonging to a tubular structure as well as certain constraints for better robustness, e.g., handling of bifurcations and low-contrast regions (Breitenreicher et al., 2013). Standard optimization strategies such as belief propagation or graph cuts are commonly applied to find the global optimum of the graph which intrinsically defines the termination criteria. However, dealing with large image data, global optimization easily becomes computationally infeasible.

Vessel Enhancement. An essential element of all algorithms are measures for tubularity or “vesselness”. They are commonly calculated based on optimal filtering and Hessian-based approaches relying on idealized appearance and noise models to enhance tubular structures. The former includes optimal edge detection (Canny, 1983) and steerable filters providing an elegant theory for computationally efficient ridge detection at arbitrary orientations (Jacob and Unser, 2004; González et al., 2009b). The latter is based on the eigenanalysis of the Hessian capturing the second-order structure of local intensity variations (Sato et al., 1997; Frangi et al., 1998). The Hessian is commonly computed by convolving the image patch with the partial second-order derivatives of a Gaussian kernel as the method of choice for noise reduction and to tune the filter response to a specific vessel scale. This basic principle has already been used by Canny for edge and line detection (Canny, 1983; Schneider, 1990). The differential operators involved in the computation of the Hessian are well-posed concepts of linear scale-space theory (Lindeberg, 1996). Modeling vessels as elongated elliptical structures, the eigendecomposition of the Hessian has a geometric interpretation, which can be used to define a vesselness measure as a function of the eigenvalues (Sato et al., 1997; Frangi et al., 1998). Due to the multi-scale nature of vascular structures, Hessian-based filters are commonly applied at different scales. Besides, the eigenvector corresponding to the largest eigenvalue of the Hessian computed at the most discriminative scale is a good estimate for the local vessel direction. In practice, vesselness filters tend to be prone to noise and have difficulty in detecting vessel parts such as bifurcations not complying with the intrinsic idealized appearance model. More recently, Xiao et al. (2013) proposed to replace the Gaussian kernel of standard Hessian approaches with a bi-Gaussian function that allows for independent selection of different scales in the foreground and background. The authors show that a proper selection of the scale parameters reduces interference from adjacent objects while preserving intra-region smoothing. As compared to Hessian-based approaches using inappropriately broad Gaussian kernels, it is hence better suited to resolve neighboring structures, in particular. Vesselness filters have also been successfully applied for global vessel segmentation in X-ray angiography using ridge tracking (Schneider and Sundar, 2010) and graph cut theory (Hernández-Vela et al., 2011). In order to partly overcome the shortcomings of Hessian-based filters, optimally oriented flux (OOF) as introduced by Law and Chung (2008) and its anisotropic variations (Benmansour and Cohen, 2011) have recently gained attention for the segmentation of different anatomical structures including vessels (Benmansour et al., 2013) and intervertebral discs (Law et al., 2013). Briefly, OOF aims at computing an optimal projection direction minimizing the inward oriented flux at the boundary of localized circles (2-D) or spheres (3-D) of different radii (scales). Similar to the Hessian-based approaches, OOF can be used to estimate the local vessel direction as a generalized eigenvalue problem. At the same time, the OOF descriptor is more robust against image noise and local intensity inhomogeneities in the presence of nearby structures, which adversely affects the differential nature of the Hessian. The OOF value, i.e., the projected outward flux, at a certain position and scale can be interpreted as the likelihood

of the voxel being centered in a tubular structure of the selected scale. By design, OOF hence provides strong responses at centerlines of curvilinear structures. Similar to the Hessian-based vesselness, the OOF eigenvalues can be combined to obtain a response across the entire structure (Law and Chung, 2008; Benmansour and Cohen, 2011). Finally, Law and Chung (2010) have demonstrated that different measures of image gradient symmetry can be derived from OOF to guide an active contour model for 3-D vessel segmentation with promising results on clinical intracranial and cardiac image data.

Centerline Extraction. For many applications, vessel detection, i.e., binary segmentation of the vessel lumen, is insufficient and a more comprehensive vascular description is desirable to characterize the topology and morphology of vascular networks. To this end, the tubular shape of a vascular segment can be modeled by its centerline, i.e., the 1-D curve centered inside the vessel lumen, along with an estimate of the vessel diameter along the centerline assuming a circular cross-section. Other centerline models rely on more general cross-sectional contours such as ellipses (Krissian et al., 2006). Various approaches for centerline extraction have been proposed in the literature including skeletonization by homotopic thinning (Palágyi and Kuba, 1998; Pudney, 1998) and minimal path techniques (Lesage et al., 2009, Section 4.4). The latter computes the centerline between two-points as the path minimizing a certain energetic potential favoring centerline positions. Minimal path techniques enjoy great popularity due to their robustness and global optimality properties (Cohen and Kimmel, 1997). Different variations have been proposed that mostly differ in the definition of the energy term and the numerical optimization schemes such as Dijkstra-like (Gülsün and Tek, 2008; Breitenreicher et al., 2013) and fast marching schemes (Sethian, 1999; Benmansour and Cohen, 2011). Deschamps (2001) defines a distance potential as a non-linear function of the distance to the object boundary. It is used to readjust minimal paths towards the vessel center. Slight inaccuracies in the extracted vessel boundaries may easily impair the distance-based metric, though. Benmansour and Cohen (2011) propose an anisotropic metric based on OOF (Law and Chung, 2008) and present promising results. However, accurate centerline extraction requires a dense sampling of the scale space which is computationally expensive when dealing with tubular structures in a wide range of scales. Recently, voting mechanisms as used for object detection in the computer vision community (Gall et al., 2011) have been applied in the context of centerline extraction to increase robustness against noise and low-contrast regions, in particular (Zhou et al., 2007; Risser et al., 2008; Rouchdy and Cohen, 2012).

1.2. Overview

In this paper, we aim at efficient processing of 3-D high-resolution angiographic image data ($> 10^{10}$ voxels) mapping the cerebrovascular system down to the capillary level, which is of great interest for the analysis of the cerebral vasculature (Reichold et al., 2009; Hirsch et al., 2012; Schneider et al., 2014). An example dataset of a small cylindrical sample of the rat somatosensory cortex

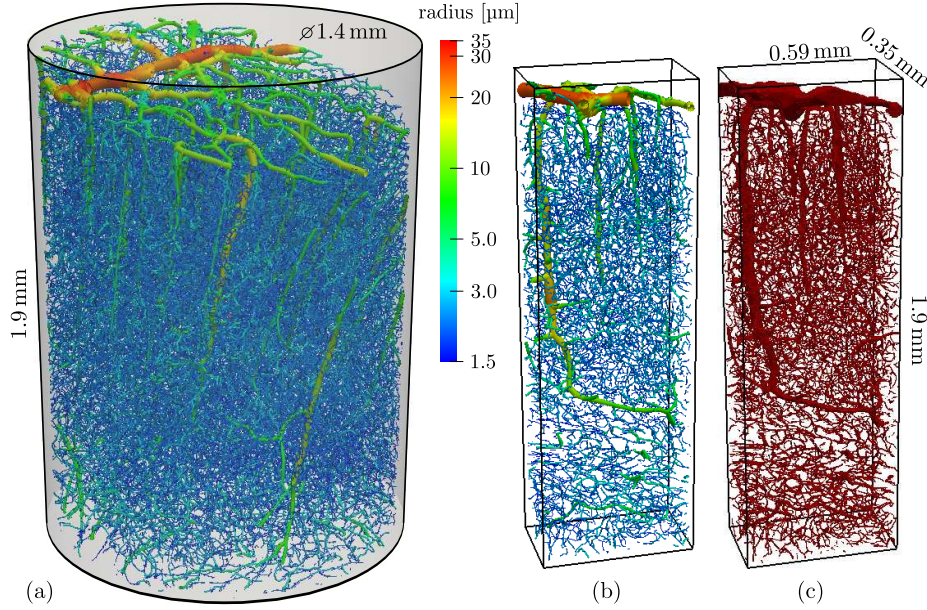


Figure 1: Cerebrovascular network of the rat somatosensory cortex. (a) Vectorized model reconstructed from 3-D high-resolution imaging data acquired by synchrotron radiation X-ray tomographic microscopy (SRXTM) of a cylindrical sample with a volume of about 2.8 mm^3 . (b,c) Cubic subregion of (a) containing a pial arteriole at the cortex surface (top) with a penetrating arteriole orthogonally plunging into the cortex. The vasculature is visualized as vectorized model (b) and surface mesh (c). The vessel radii in (a) and (b) are color-coded on a logarithmic scale. [Best viewed in color]

is shown in Figure 1. Processing these huge amounts of data involves various challenges: First and foremost, the considerable computational challenge due to the *gigantic data size* – more than 20 GB of raw image data for the small sample in Figure 1a – requires particularly efficient algorithms to perform the analysis within a reasonable time span. We hence devise a fast machine learning approach for joint vessel segmentation and centerline extraction using a single set of features computed from efficient local linear filtering rather than complex non-local spatial models incorporating prior knowledge and regularization (Schneider and Sundar, 2010; Hernández-Vela et al., 2011). Second, manual or semi-automatic generation of high quality ground truth data by human experts is tedious and essentially prohibitive for high-resolution 3-D data. Training the models of machine learning techniques, however, requires sufficient amount of labeled training data. Therefore, *minimum manual input* during training is another crucial objective of this work. To this end, we propose to train the classifiers using automatically generated noisy training labels, thus eliminating the need for tedious manual labeling. Despite typical imaging artifacts and high levels of label noise in the training data, the obtained segmentation results prove robust and accurate on our datasets. Finally, vascular structures are characterized by a *complex topology and morphology* with large variations, which adds to the complexity of automated

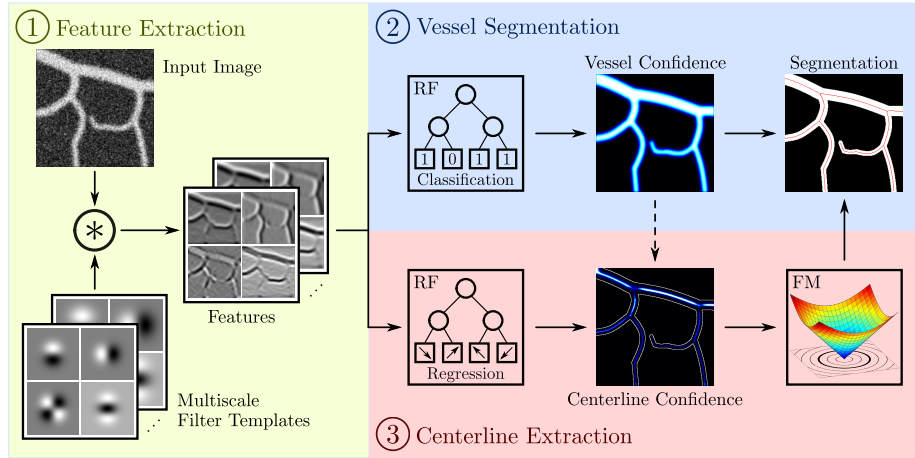


Figure 2: Overview of the machine learning-based framework for vessel segmentation and centerline extraction for a 2-D example. First, features are computed from (steerable) filter templates at different scales (left). Second, the vessel lumen is segmented based on random forest (RF) classification (top right). Third, we cast probabilistic votes for the centerline location at each location labeled as vessel using multivariate Hough forest regression (bottom right). The centerline is finally extracted using a fast marching (FM) framework.

vessel segmentation. Due to the hierarchical architecture of the vascular system, vessels appear at very different scales spanning several orders of magnitude. Considering this complex multi-scale nature of vascular structures, we implicitly learn the vascular appearance from the training data employing oriented features computed at different scales. The learned model is more flexible and superior to parametric models, e.g., based on Hessian eigenanalysis (Sato et al., 1997; Frangi et al., 1998) or optimally oriented flux (Law and Chung, 2010), that intrinsically rely on a cylindrical appearance model which clearly does not hold in special topological configurations such as bifurcations. Figure 2 provides an overview of the proposed processing pipeline.

1.3. Contributions

In the following we briefly summarize the main contributions of this work¹:

1. We propose a novel framework for joint vessel segmentation and centerline extraction. This framework predicts – simultaneously and from the same image features – local vessel presence and nearby centerlines using, respectively, oblique random forests (RFs) for efficient classification and multivariate Hough forests to infer probabilistic votes about the supposable vessel center. The accumulated centerline votes then drive a fast march-

¹Contributions partly included in our previous conference paper (Schneider et al., 2013) are marked with an asterisk.

ing (FM) scheme to extract the centerline as the minimal path with lowest energy. We refer to this framework as oblique Hough forest.

- 2.* Based on extensive validation experiments, we offer a systematic comparative analysis of different features for vessel segmentation computed from, respectively, orthogonal subspace filtering (Menze et al., 2006; Rigamonti et al., 2011) and steerable filters sharing similar structure (Freeman and Adelson, 1991; Jacob and Unser, 2004). The latter allows for efficient directional filtering and explicit scale parametrization in order to compensate for the preferred local orientation and to account for the multi-scale nature of vascular structures. Steerable filters have successfully been applied for filament and dendrite detection in previous work by González et al. (2009a,b).
- 3.* We further propose and comprehensively test a novel oblique split model with an elastic net penalty term that combines ℓ_1 and ℓ_2 regularization, which leads to sparser split weights than purely ℓ_2 regularized oblique splits introduced by Menze et al. (2011).
4. In order to minimize the requirement for manual input during training, we use automatically generated noisy annotations to train the classification and regression forests in our validation experiments. In a systematic quantitative assessment of the parametrization of the RF models we demonstrate the advantage of the learning step and show that the trained classification and regression forests are able to generate and extract highly accurate segmentation results and centerlines in spite of the high level of label noise in the training data.

1.4. Paper Outline

This paper is structured as follows. In Section 2, we first introduce two different kinds of features based on orthogonal subspace filtering and steerable filter templates, respectively, that are used for vessel segmentation. Second, we present the details of our machine learning framework based on random forests, especially focusing on the design of the split functions. We also show how the framework can be extended to allow for centerline extraction by applying a fast marching (FM) approach to a “centeredness” potential obtained from Hough forest regression. In Section 3, we give details about the performed validation and sensitivity experiments to assess the segmentation performance and accuracy of our framework. The results are discussed in Section 4. Finally, we conclude and show perspectives in Section 5.

2. Methods

In this section, we first introduce two different sets of local image features for vessel segmentation based on (1) orthogonal subspace filtering and (2) steerable filters computed at different scales and orientations in order to achieve rotational

invariance (Section 2.1). In Section 2.2, these features are then used to train an oblique RF classifier that is well adapted to correlated feature responses from local image filters (Menze et al., 2011). Different from standard discriminative learning algorithms, such as support vector machines, RF naturally returns continuous probabilities when predicting vessel locations, which allows us to adapt the threshold for classification. Moreover, RF is capable of coping with high dimensional feature vectors and tolerate noisy training labels. It is fast to train with only very few parameters to be optimized, even faster to apply (few threshold operations), and easy to parallelize. Efficient prediction becomes particularly important for scalability in our specific application using high-resolution image data at nanometer resolution. Finally, Section 2.3 describes an extension of the RF classification framework that allows for robust and accurate centerline extraction using a Hough voting approach based on regression forests.

2.1. Local Image Features

We introduce two different kinds of features. One is based on orthogonal subspace filtering where we learn 3-D eigenspace filters from local image patches that return task optimal feature responses. The other uses a set of steerable filters that show, qualitatively, similarities to the learned eigenspace filters, but also allow for explicit parametrization of scale and orientation that we formally generalize to the 3-D spatial context. In this way, steerable filters allow for efficient computation of oriented features along arbitrary directions in 3-D and at the same time include higher order derivatives as compared to Hessian-based approaches.

2.1.1. Orthogonal Subspace Filters (OSFs)

Matched filters (MFs) have widely been used in signal processing. They allow to detect a signal of known shape (template) by cross-correlation and perform provably optimal under additive Gaussian white noise conditions (Moon and Stirling, 2000). In terms of image processing, this corresponds to the convolution of the image with the MF. From a learning and classification perspective, matched filtering (signal detection) is closely related to linear regression for binary classification between background and pattern (vessel) (Menze et al., 2006). Considering the image as a composition of local image patches with each pixel in the patch representing a feature, MF defines a 1-D linear subspace (regression coefficients) of this feature space which allows for separation of the pattern from background. Instead of an optimal 1-D subspace assuming linear separability in the feature space as implied by using a single matched filter, we use a less restrictive dimensionality reduction similar to (Menze et al., 2006), namely (linear) principal component analysis (PCA), in order to define a subspace of higher dimensionality. More formally, let $\mathbf{p}_i \in \mathbb{R}^{P^3}$ denote a (cubic) image patch of size $P \times P \times P$ containing P^3 pixels. A d -dimensional subspace ($d \leq P^3$) capturing the most important modes of variation in the image patches

can then be defined using PCA (Jolliffe, 2002):

$$\forall 1 \leq k \leq d \leq P^3 : \boldsymbol{\alpha}_k = \arg \max_{\substack{\boldsymbol{\alpha} \in \mathbb{R}^{P^3}, \|\boldsymbol{\alpha}\|=1, \\ \forall 1 \leq i < k: \text{cov}(\boldsymbol{\alpha}_i, \boldsymbol{\alpha})=0}} \text{var}(\boldsymbol{\alpha}^\top P_{\text{OSF}}) \quad , \quad (1)$$

where $P_{\text{OSF}} = [\mathbf{p}_i]_{1 \leq i \leq N_P} \in \mathbb{R}^{P^3 \times N_P}$ is the data matrix assembling N_P patches labeled as vessel. The principal axes $\boldsymbol{\alpha}_k$ form an orthonormal basis of the d -dimensional subspace and are ordered according to their preserved variance. They can be computed efficiently as the d eigenvectors corresponding to the largest eigenvalues of the covariance matrix of P_{OSF} after mean centering using singular value decomposition. Projecting an arbitrary image patch $\mathbf{p} \in \mathbb{R}^{P^3}$ onto the PCA subspace yields its d principal components (PCs) $\boldsymbol{\alpha}_k^\top (\mathbf{p} - \mathbf{m}_1)$, where \mathbf{m}_1 is the empirical mean of all patches in P_{OSF} . The PCs of the image patches centered at pixels \mathbf{x} in image I can thus be computed by d independent convolution operations of the image with each (properly reshaped) principal axis $\tilde{\boldsymbol{\alpha}}_k \in \mathbb{R}^{P \times P \times P}$:

$$\mathbf{f}_{\text{OSF}}(I, \mathbf{x}) = \left[(\tilde{\boldsymbol{\alpha}}_k * I)(\mathbf{x}) - \boldsymbol{\alpha}_k^\top \frac{1}{N_P} \sum_{i=1}^{N_P} \mathbf{p}_i \right]_{1 \leq k \leq d} \in \mathbb{R}^d \quad . \quad (2)$$

The (reshaped) principal axes will also be referred to as orthogonal subspace filters (OSFs). The PCs, i.e., the OSF response of an image patch, are used as features along with a non-linear decision rule for vessel segmentation as described in Section 2.2.

Rotational Invariance. In the definition of the OSFs in Equation (1), we rely on the PCA-based scheme to learn the structure of the high-dimensional patch feature space and to provide a proper feature projection into a low-dimensional subspace. Instead of learning the OSFs from the raw image patches, we can apply the same scheme to local patches that have been normalized for pose in order to computationally eliminate the variability in the orientation of vascular structures. To this end, each image patch \mathbf{p}_i has to be transformed w.r.t. the local vessel direction into a normalized pose first (see Section 2.1.2 for details), which requires costly resampling and interpolation. Apart from patch normalization, the framework for learning the rotationally invariant orthogonal subspace filters (rOSFs) and the extraction of rOSF features is the same as for OSF (see Equations (1) and (2)). While rOSF is prohibitively expensive for most applications – including ours – we report experimental results for both OSF and rOSF features for validation purposes in Section 3.

2.1.2. Steerable Filter Templates (SFTs)

The OSFs learned from image patches as described in the previous section turn out to be highly structured (see Figure 6). Instead of learning the structured filter kernels, we attempt to explicitly parametrize them using a steerable filter model. The model we choose is based on Gaussian derivatives, which allows for

efficient directional filtering at different scales and, most importantly, implicates rotational invariance (Jacob and Unser, 2004). Similar to González et al. (2009b), we define the filter templates as normalized derivatives of Gaussians up to order M (Lindeberg, 1996):

$$\forall m \geq 1 \wedge 0 \leq b \leq a \leq m \leq M : G_{m,a,b}^\sigma(\mathbf{x}) = \sigma^m \frac{\partial^{m-a} \partial^{a-b} \partial^b}{\partial x^{m-a} \partial y^{a-b} \partial z^b} G^\sigma(\mathbf{x}) \quad , \quad (3)$$

where $G^\sigma(\mathbf{x}) = \frac{1}{(\sqrt{2\pi}\sigma)^3} \exp(-\frac{\|\mathbf{x}\|^2}{2\sigma^2})$ denotes the 3-D symmetric Gaussian kernel with variance σ and zero mean. As in Equation (2), each template induces a single feature by convolution with image I . They can be assembled to a feature vector of dimension $d_M = \frac{1}{6}(M^3 + 6M^2 + 11M)$ at a fixed scale σ :

$$\mathbf{f}^\sigma(I, \mathbf{x}) = ((G_{1,0,0}^\sigma, G_{1,1,0}^\sigma, G_{1,1,1}^\sigma, \dots, G_{M,M,M}^\sigma)^\top * I)(\mathbf{x}) \in \mathbb{R}^{d_M} \quad . \quad (4)$$

We enhance the features by concatenating feature vectors at different scales $\boldsymbol{\sigma} = (\sigma_1, \dots, \sigma_S)^\top$:

$$\mathbf{f}_{\text{SFT}}(I, \mathbf{x}; \boldsymbol{\sigma}) = (\mathbf{f}^{\sigma_1}(I, \mathbf{x}), \dots, \mathbf{f}^{\sigma_S}(I, \mathbf{x}))^\top \in \mathbb{R}^{d_M S} \quad . \quad (5)$$

The steerability of Gaussian derivatives has been derived for the 2-D case in (Jacob and Unser, 2004) and can readily be extended to 3-D (Freeman and Adelson, 1991; González et al., 2009a). Steerability refers to the property that the convolution of an image with a rotated version of the steerable filter template (SFT) can be expressed by a linear combination of the filter response of the image with the SFT without rotation:

$$I * G_{m,a,b}^\sigma(R\mathbf{x}) = \sum_{i=0}^m \sum_{j=0}^i \omega_{m,a,b}^{i,j} \underbrace{(I * G_{m,i,j}^\sigma)(\mathbf{x})}_{\mathbf{f}_{m,i,j}^\sigma(I, \mathbf{x})} \quad , \quad (6)$$

where $R \in SO(3)$ denotes a 3-D rotation matrix. The uniquely defined coefficients $\omega_{m,a,b}^{i,j}$ can be computed in closed form as:

$$\begin{aligned} w_{m,a,b}^{i,j} = \sum_{\mathbf{p} \in \mathcal{P}_{m,a,b}^{i,j}} & (-1)^{a-v_1-w_2} \binom{m-a}{u_1} \binom{a-b}{v_1} \binom{b}{w_1} \binom{u_1}{u_2} \binom{w_1}{w_2} \\ & (\cos \theta)^{m-a-u_2+w_2} (\cos \phi)^{m-a+b-u_1+v_1-w_1} \\ & (\sin \theta)^{b+u_2-w_2} (\sin \phi)^{a-b+u_1-v_1+w_1-u_2-w_2} \quad , \end{aligned} \quad (7)$$

where

$$\begin{aligned} \mathcal{P}_{m,a,b}^{i,j} = \{ & (u_1, v_1, w_1, u_2, w_2)^\top \in \mathbb{N}_0^5 \mid u_1 \leq m-a, v_1 \leq a-b, \\ & w_1 \leq b, u_2 \leq u_1, w_2 \leq w_1, \\ & u_1 + v_1 + w_1 = i, u_2 + w_2 = j \} \quad . \end{aligned} \quad (8)$$

A more detailed derivation is provided in Appendix B. This formalism allows to efficiently evaluate the feature vector \mathbf{f}_{SFT} for an arbitrary rotation without any additional costly convolution. We use a restricted set of rotations in our application considering the tubular structure of vessels. The local vessel direction $\mathbf{d} = (d_x, d_y, d_z)^\top \in \mathbb{R}^3$, $\|\mathbf{d}\| = 1$ can be parametrized using spherical coordinates (θ, ϕ) with unit radius, elevation $\theta = \arctan(d_z / \sqrt{d_x^2 + d_y^2})$, and azimuth $\phi = \arctan(d_y / d_x)$ relative to the x - y plane ($z = 0$). It is sufficient to restrict the parametrization to the positive hemisphere ($z > 0$), i.e., $0 \leq \theta \leq \pi/2$ and $-\pi < \phi \leq \pi$. The vessel patch can then be transformed to the normalized pose $\mathbf{d}_0 = (1, 0, 0)^\top$ by applying the rotation matrix

$$R_{\theta, \phi} = \begin{pmatrix} \cos \theta \cos \phi & \cos \theta \sin \phi & \sin \theta \\ -\sin \phi & \cos \phi & 0 \\ -\sin \theta \cos \phi & -\sin \theta \sin \phi & \cos \theta \end{pmatrix}. \quad (9)$$

The SFT features evaluated for this rotation according to Equation (6) hence describe the intensity variation characteristics of different orders along the vascular structure as well as in the orthogonal plane. Assuming a symmetric vessel (intensity) profile perpendicular to the local vessel direction \mathbf{d} , restricting the set of rotations is reasonable as the vessel structure is (locally) invariant under rotation about \mathbf{d} .

Features computed from SFT have several advantages over (r)OSF: First of all, the explicit scale parameter σ allows to tune the SFT features to better account for the multi-scale nature of vascular structures. Second, SFT features can be computed efficiently for arbitrary (vessel) orientations due to the steerability of Gaussian derivatives. Directional filtering based on rOSF, by contrast, involves computationally costly resampling and interpolation due to the lack of steerability (see Section 2.1.1). While this is still feasible for the estimation of the rOSF filter templates, it becomes extremely costly during prediction particularly when maximizing the ensemble confidence for different orientations (see Section 2.2). Finally, the separability of Gaussian derivatives dramatically decreases the computational complexity of SFT as compared to the inseparable (r)OSF filters, which is of uttermost importance for the processing of large image volumes as in our application.

2.2. Vessel Segmentation Using Classification Forests

The OSF and SFT features as defined in Equations (2) and (5), respectively, are each used along with a non-linear decision rule for vessel segmentation. We train separate classifiers for the different feature types as follows:

2.2.1. Training Data

A representative set S of $2N_S$ tuples (image I_k , location \mathbf{x}_k , vessel orientation \mathbf{d}_k , class label y_k) is randomly sampled from a labeled set of images corresponding to N_S foreground ($y_k = 1$) and background ($y_k = 0$) samples, respectively:

$$S = \{(I_k, \mathbf{x}_k, \mathbf{d}_k) \mid 1 \leq k \leq 2N_S \wedge y_k = 1 \iff 1 \leq k \leq N_S\} \quad (10)$$

For these samples, the features $\mathbf{f}(I, \mathbf{x})$ can be extracted as defined in Equations (2) and (5). The SFT features are additionally rotated to the normalized orientation according to Equations (6) and (9) w.r.t. the local vessel direction \mathbf{d} . Similarly, normalized rOSF features are obtained from rotated (and resampled) image patches. This defines the training set

$$\mathcal{T}^{(0)} = \{(\mathbf{f}_{\mathbf{k}} = \mathbf{f}(I_k, \mathbf{x}_{\mathbf{k}}), y_k) \mid 1 \leq k \leq 2N_S\} \quad (11)$$

that is ultimately used to train a random forest (RF) classifier (Breiman, 2001).

2.2.2. Training

An RF classifier consists of an ensemble of decision trees used to model the posterior probability of each class (vessel/background). During training, each tree is fully grown from bootstrapped datasets² $\mathcal{T}' \subset \mathcal{T}^{(0)}$ using stochastic discrimination. For this, the data is split at each tree node, starting at the root, by a binary test defining a hyperplane in the feature (sub-)space. In contrast to traditional bagging, the split is based on a small number of randomly selected feature channels c_1, \dots, c_{N_F} only

$$\Lambda(\mathbf{f}; \mathbf{c}, \mathbf{w}, \theta) = \begin{cases} 0, & \text{if } \mathbf{w}^\top [\mathbf{f}_{c_i}]_{i=1, \dots, N_F} < \theta \\ 1, & \text{otherwise} \end{cases}, \quad (12)$$

where $\mathbf{w} \in \mathbb{R}^{N_F}$ and $\theta \in \mathbb{R}$ are the split parameters to be determined during training. Starting at the root node, each training sample is split w.r.t. the chosen binary test and serves as input for the construction of the left ($\Lambda = 0$) or right ($\Lambda = 1$) child node. The construction process is repeated recursively until a break criterion is met, i.e., the maximum tree depth has been reached or the number of training samples available to estimate another split has dropped below a certain threshold. Upon termination at a leaf node l , the remaining training samples $\mathcal{T}_l \subset \mathcal{T}^{(0)}$ are used to compute the empirical estimate of the posterior class probabilities as

$$\begin{aligned} p(y = 1 | \mathcal{X}_l) &= \frac{1}{|\mathcal{X}_l|} \sum_{k=1}^{|\mathcal{X}_l|} y_k, \\ p(y = 0 | \mathcal{X}_l) &= 1 - p(y = 1 | \mathcal{X}_l). \end{aligned} \quad (13)$$

Note that for a given training (sub-)set, we refer to the corresponding (ordered) multiset of features and labels as \mathcal{X} and \mathcal{Y} , respectively.

We investigated both “orthogonal” and “oblique” trees varying in the applied split model. As proposed in Breiman’s original paper (Breiman, 2001), the former is based on optimal thresholds for randomly selected single features in every split ($N_F = 1$, $\mathbf{w} = 1$), i.e., mutually orthogonal 1-D hyperplanes. The

²We stick to the “set” terminology and notation here even though, strictly speaking, bootstrapping yields (ordered) multisets denoted by capital calligraphic letters.

computation of the optimal split parameters during the initial training phase is driven by maximizing the information gain, i.e., minimizing the class uncertainty

$$\begin{aligned}\hat{\Lambda} &= \arg \max_{\Lambda} H(\mathcal{X}) - \sum_{k \in \{0,1\}} \frac{|\mathcal{X}_k|}{|\mathcal{X}|} H(\mathcal{X}_k) \\ &= \arg \min_{\Lambda} \underbrace{\sum_{k \in \{0,1\}} -|\mathcal{X}_k| \sum_{y \in \{0,1\}} p(y|\mathcal{X}_k) \log(p(y|\mathcal{X}_k))}_{U_k(\mathcal{T}; \Lambda) = U_k(\mathcal{X}, \mathcal{Y}; \Lambda)} \quad ,\end{aligned}\quad (14)$$

where H is the entropy and $\mathcal{X}_k = \{\mathbf{f} \in \mathcal{X} \mid \Lambda(\mathbf{f}) = k\}$.

By contrast, oblique splits define multidimensional hyperplanes to separate the feature space, e.g., by choosing randomly oriented hyperplanes (Breiman, 2001) or applying linear discriminative models (Menze et al., 2011). For the oblique RFs in this work, we employ a linear regression model with an elastic net penalty term (Friedman et al., 2010) to learn multivariate (optimal) split directions \mathbf{w} at each node:

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w} \in \mathbb{R}^{N_F}} \frac{1}{2|\mathcal{T}_n|} \sum_{(\mathbf{f}, y) \in \mathcal{T}_n} \left((2y - 1) - \mathbf{w}^\top [\mathbf{f}_{c_i}]_{i=1, \dots, N_F} \right)^2 + \lambda P_\alpha(\mathbf{w}) \quad , \quad (15)$$

where $\mathcal{T}_n \subset \mathcal{T}^{(0)}$ is the training data passed to node n and c_1, \dots, c_{N_F} are randomly selected (but fixed) feature channels. Note that we map the $(0, 1)$ -class labels y to $(-1, 1)$. As a regularizer, we employ the elastic net penalty

$$P_\alpha(\mathbf{w}) = (1 - \alpha) \frac{1}{2} \|\mathbf{w}\|_{\ell_2}^2 + \alpha \|\mathbf{w}\|_{\ell_1} \quad , \quad (16)$$

with regularization parameter $\alpha > 0$ as a compromise between ridge regression ($\alpha = 0$) and the lasso penalty ($\alpha = 1$), where $\|\cdot\|_{\ell_1}$ and $\|\cdot\|_{\ell_2}$ denote the ℓ_1 - and ℓ_2 -norm, respectively. The advantage is joint regularization of the coefficients and sparsity – coefficients are both encouraged to be small, and to be zero if they are very small. The latter lasso property reduces the dimensionality of the split space, which is desirable for memory and robustness purposes. With $\alpha = 1$ (and $\lambda \gg 0$) we will get a single nonzero coefficient, i.e., RF with univariate splits, whereas choosing $\alpha = 0$ we have ridge regression as in (Menze et al., 2011). Altogether, we train a random forest classifier by separately growing N_T decision trees as summarized in Algorithm 1.

2.2.3. Prediction

Previously unseen samples \mathbf{x} can be classified by pushing the extracted features \mathbf{f} down all N_T decision trees of the ensemble until it ends up in a leaf node. Thus, each tree assigns a label $\hat{y}_k \in \{0, 1\}$ corresponding to the maximum posterior class probability stored at the leaf node. Similar to Equation (13), the combined posterior is estimated as the ensemble confidence:

$$L_V(\mathbf{x}) = p(y = 1|\mathbf{f}) = \frac{1}{N_T} \sum_{k=1}^{N_T} \hat{y}_k \quad . \quad (17)$$

Algorithm 1 Train RF Classifier

```

1: for  $t \leftarrow 1$  to  $N_T$  do ▷ grow single tree
2:    $\mathcal{T}' \leftarrow$  set of  $\frac{2}{3}|\mathcal{T}^{(0)}|$  bootstrapped samples from training data  $\mathcal{T}^{(0)}$ 
3:    $\text{stack.push}((n = n_{\text{root}}, d = 0, \mathcal{T} = \mathcal{T}'))$  ▷ start at root node, depth  $d = 0$ 
4:   while stack is not empty do
5:      $(n, d_n, \mathcal{T}) \leftarrow \text{stack.pop}()$  ▷ next node/training data to be processed
6:     for  $s \leftarrow 1$  to  $\lceil \sqrt{N_F^0} \rceil$  do ▷ estimate optimal split candidates
7:        $\mathbf{c}_s = (c_1, \dots, c_{N_F}) \leftarrow$  randomly sampled feature channels without
          replacement from feature pool of size  $N_F^0$ 
8:       switch Split node model do ▷ estimation of split parameters
9:         case Orthogonal split ( $N_F = 1$ )
10:          Compute optimal split threshold w.r.t. Equation (14) from
            selected feature channel  $c_1$  over all samples
11:           $\hat{\theta} \leftarrow \arg \min_{\mathbf{x} \in \mathcal{X}_n} \sum_{k \in \{0,1\}} U_k(\mathcal{T}_n; \Lambda(\mathbf{f}; \mathbf{c} = c_1, \mathbf{w} = 1, \theta = \mathbf{x}_c))$ 
12:         case Oblique split ( $N_F = \lceil \sqrt{N_F^0} \rceil$ )
13:          Normalize selected features of bootstrapped samples to zero
            mean and unit variance to enhance the stability of the linear
            model
14:           $\hat{\mathbf{w}} \leftarrow$  Optimal split direction computed from Equation (15)
            using covariance updates (Friedman et al., 2010) for
             $\alpha = 0.5$  (see Section 4.3 for details on the choice of  $\lambda$ 
            using pathwise coordinate descent)
15:           $\hat{\theta} \leftarrow$  optimal bias maximizing the information gain (see Equa-
            tion (14))
16:           $\Lambda_s \leftarrow \Lambda(\mathbf{f}; \hat{\mathbf{w}}, \hat{\theta})$  ▷ split candidate
17:        end for
18:         $\hat{\Lambda} \leftarrow$  optimal split of all split candidates  $\Lambda_s$  maximizing the information
          gain (see Equation (14))
19:        Store optimal split  $\hat{\Lambda}$  and selected feature channels  $\mathbf{c}_s$  at tree node  $n$ 
20:        if  $d < d_{\text{max}}$  and  $|\mathcal{T}| < s_{\text{min}}$  then ▷ maximum tree depth reached or there are
          too few training samples available to ro-
          bustly estimate another split
21:          Compute empirical posterior class probabilities based on  $\mathcal{T}$  (see
            Equation (13)) at and store it at current leaf node  $n$ 
22:        else
23:           $(\mathcal{T}_L, \mathcal{T}_R) \leftarrow$  split training data  $\mathcal{T}$  w.r.t. optimal split function  $\hat{\Lambda}$ 
24:          Create left and right child nodes for node  $n$ 
25:           $\text{queue.push}((n_L, d_n + 1, \mathcal{T}_L))$  ▷ left child node
26:           $\text{queue.push}((n_R, d_n + 1, \mathcal{T}_R))$  ▷ right child node
27:        end if
28:      end while
29:    end for

```

The binary class label \hat{y} can finally be assigned using a majority vote, i.e., $L_V(\mathbf{x}) > 0.5$, or any other threshold.

Note that for OSF features a single RF is trained for all vessel orientations. The intrinsic orientation-induced structure in the OSF feature space has to be learned by arbitrarily rotating image patches both for learning the OSF eigenfilters and for training the subsequent classifier (see Section 3 for further details). In contrast, SFT features allow for explicit parametrization of the orientation. The expected filter response for an arbitrary orientation can efficiently be computed from the set of stationary base features \mathbf{f}_{SFT} as defined in Equations (5) and (6). As the corresponding RF classifiers are trained on SFT features extracted from vessels with normalized orientation only, the SFT features of the patch to be classified have to be computed for normalized orientation as well.³ To this end, orientation independent prediction can be achieved by sampling the space of possible vessel orientations (half sphere), computing the corresponding (rotated) SFT features and ultimately assigning the classification result with the maximum confidence as proposed in (González et al., 2009b). In contrast to OSF features, this even allows to not only estimate the posterior class probabilities but also a probability distribution on the vessel orientation. Using a robust predictor for the local vessel direction, e.g., based on the eigenanalysis of the Hessian, the prediction stage can certainly be sped up substantially.

2.3. Centerline Extraction Using Multivariate Hough Regression Forests

In this section, we describe how to extend the segmentation framework to allow for joint vessel segmentation and centerline extraction using a fast marching approach applied to an energy potential obtained from a probabilistic regression framework. To this end, we train a Hough regression forest, i.e., an ensemble of regression trees, to infer at each voxel labeled as vessel the most likely displacements to proximate vessel centers. Regression forests enjoy great popularity in the computer vision community for object detection (Gall and Lempitsky, 2009; Rematas and Leibe, 2011). To our knowledge, the only application of regression forests in the medical field is for anatomy detection and localization by Criminisi et al. (2011).

2.3.1. Centerline Training Data

The construction of the regression trees proceeds very similar to the RF training described in the previous Section 2.2. For the definition of the training set, instead of a binary class label, we now assign to each image location \mathbf{x}_k the offset $\mathbf{o}_k = \mathbf{c}_k - \mathbf{x}_k \in \mathbb{R}^3$ to the “closest” location $\mathbf{c}_k \in C$ along the vessel

³Note that this also holds for rOSF features.

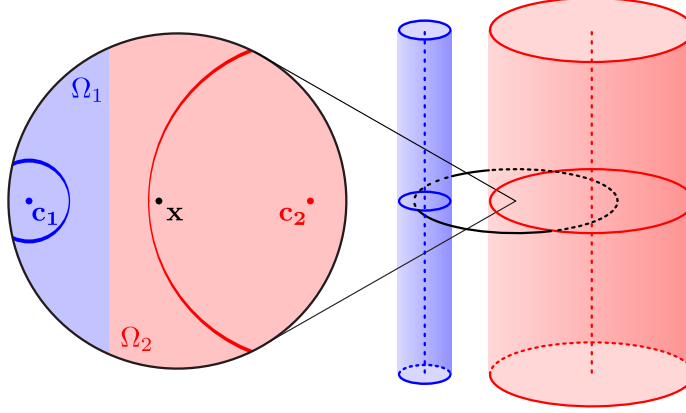


Figure 3: Visualization of the mapping from image locations to the “closest” point along the centerline for two cylindrical vessel segments with their centerlines marked as dashed lines (right). The mapping as defined in Equation (18) is visualized for an enhanced axial subregion (left). The image subdomains Ω_1 and Ω_2 shaded in blue and red, respectively, are assigned to the corresponding centerline location $\hat{\mathbf{c}}(\Omega_i) = \mathbf{c}_i$ ($i \in \{1, 2\}$). For the image location \mathbf{x} , in particular, \mathbf{c}_1 is the closest centerline point w.r.t. ℓ_2 as $\|\mathbf{x} - \mathbf{c}_1\| < \|\mathbf{x} - \mathbf{c}_2\|$, whereas $\hat{\mathbf{c}}(\mathbf{x}) = \mathbf{c}_2$ yields the topologically correct mapping.

centerline C :

$$\hat{\mathbf{c}}(\mathbf{x}) = \begin{cases} \arg \min_{\substack{\mathbf{c} \in C \\ Y(\mathbf{x} - t(\mathbf{x} - \mathbf{c})) = 1 \ \forall t \in [0, 1]}} \|\mathbf{x} - \mathbf{c}\|, & \text{if } Y(\mathbf{x}) = 1 \\ \hat{\mathbf{c}}(\arg \min_{\substack{\mathbf{x}' \in \Omega \\ Y(\mathbf{x}') = 1}} \|\mathbf{x} - \mathbf{x}'\|), & \text{otherwise} \end{cases}, \quad (18)$$

where Ω is the image domain. The binary segmentation map Y assigns a binary label to each location depending on whether the location belongs to a vessel ($Y = 1$) or not ($Y = 0$). In other words, a voxel labeled as vessel is assigned to the closest point on the centerline such that the straight path between the point and the centerline candidate is completely contained in the lumen. Each background voxel is mapped to the centerline location of the closest point of the vessel (surface). Simply mapping each image location to the closest centerline point w.r.t. ℓ_2 would result in topologically incorrect mappings (see Figure 3 for an example). The training set for centerline regression can then be defined as $\tilde{\mathcal{T}}^{(0)} = \{(\mathbf{f}_k = \mathbf{f}(I_k, \mathbf{x}_k), \mathbf{o}_k = \hat{\mathbf{c}}(\mathbf{x}_k) - \mathbf{x}_k) \mid 1 \leq k \leq N_S\}$ with the same features \mathbf{f} as used for classification. The (ordered) multiset of offset vectors \mathbf{o}_k will be referred to as \mathcal{O} .

2.3.2. Training

For the construction of the regression trees, optimal split parameters are computed by linear regression similar to Equations (14) and (15) minimizing the

uncertainty (variance) of the displacement vectors (Gall et al., 2011):

$$\hat{\Lambda} = \arg \min_{\Lambda} \sum_{k \in \{0,1\}} \tilde{U}_k(\tilde{\mathcal{T}}; \Lambda) + \lambda P_{\alpha}(\mathbf{w}) \quad , \quad (19)$$

with the uncertainty defined as:

$$\tilde{U}_k(\tilde{\mathcal{T}}; \Lambda) = |\tilde{\mathcal{O}}_k| \text{var}(\tilde{\mathcal{O}}_k) = \sum_{\mathbf{o} \in \tilde{\mathcal{O}}_k} \left\| \mathbf{o} - \frac{1}{|\tilde{\mathcal{O}}_k|} \sum_{\mathbf{o}' \in \tilde{\mathcal{O}}_k} \mathbf{o}' \right\|^2 \quad , \quad (20)$$

where $\tilde{\mathcal{O}}_k = \{\mathbf{o}_1 \in \mathcal{O} \mid \exists(\mathbf{f}_1, \mathbf{o}_1) \in \tilde{\mathcal{T}} : \Lambda(\mathbf{f}_1) = k\}$. Once the break criterion for the iterative tree construction is met (see Algorithm 1), a leaf node l is added to the tree storing the corresponding set of displacement vectors \mathcal{O}_l .

2.3.3. Prediction

During prediction, similar to the classification case, the extracted features \mathbf{f} of a previously unseen sample \mathbf{x} are pushed down each regression tree of the ensemble. Let $L(\mathbf{f})$ denote the corresponding set of reached leaf nodes. The posterior probability for the inferred location of the vessel centerline \mathbf{c} can be decomposed as

$$p(\mathbf{c}|\mathbf{x}) = \sum_{l \in L(\mathbf{f})} p(\mathbf{c}|\mathbf{x}, l) p(l) \quad . \quad (21)$$

Following Gall et al. (2011), the first term is approximated by a sum of Dirac measures $\delta_{\mathbf{o}}$ for the displacement vectors \mathbf{o} :

$$p(\mathbf{c}|\mathbf{x}, l) = \frac{1}{|\mathcal{O}_l|} \sum_{\mathbf{o} \in \mathcal{O}_l} \delta_{\mathbf{o}}(\mathbf{c} - \mathbf{x}) \quad . \quad (22)$$

Put differently, this can be interpreted as casting votes for centerline candidates located at $\mathbf{c} = \mathbf{x} + \mathbf{o}$ for each displacement $\mathbf{o} \in \mathcal{O}_l$ stored at the leaf node l . The probability $p(l)$ is used to weight the different votes w.r.t. the uncertainty of the corresponding leaves (see Equation (20)):

$$p(l) = \frac{\exp(-\eta \text{var}(\mathcal{O}_l))}{\sum_{l' \in L(\mathbf{f})} \exp(-\eta \text{var}(\mathcal{O}_{l'}))} \quad , \quad (23)$$

where $\eta \geq 0$ is a constant shape parameter. Accumulating the displacement votes over the entire image domain Ω finally yields a confidence map indicating the likelihood of an image location $\mathbf{c} \in \Omega$ being part of the vessel centerline:

$$L_C(\mathbf{c}) = \sum_{\mathbf{x} \in \Omega} p(\mathbf{c}|\mathbf{x}) p(\mathbf{x}) \quad , \quad (24)$$

with the prior defined as

$$\begin{aligned} p(\mathbf{x}) &= \sum_{y \in \{0,1\}} p(\mathbf{x}|Y(\mathbf{x}) = y) p(Y(\mathbf{x}) = y) \\ &\propto \delta_{Y(\mathbf{x})} L_V(\mathbf{x}) \quad , \end{aligned} \quad (25)$$

where L_V denotes the vessel likelihood from Equation (17). It is preferable to accumulate the votes from samples \mathbf{x} belonging to the vasculature ($Y(\mathbf{x}) = 1$), for two reasons: First, the utilized features have a rather limited spatial support depending on the scale parameters (see Equations (2) and (5)). Therefore, casting centerline votes is only meaningful at locations where the local feature descriptor captures sufficient information about the foreground structure in the local vicinity to allow for reliable inference of meaningful displacement estimates. This could partly be resolved by, e.g., proper modification of the leaf prior (see Equation (23)) or additional conditioning of the per-leaf centerline likelihood (see Equation (22)) on the class label $Y(\mathbf{x})$. Second, it remains unclear how to consistently orient the local feature descriptors at background samples to be used for centerline regression. For instance, one could choose the orientation of the “closest” vessel, which is difficult to estimate robustly, though. Alternatively, the choice of the training samples could account for this by incorporating features computed at background samples for various different orientations. This requires excessive sampling and hence significantly extends the training (and testing) phase. RF training could also be modified to incorporate the class label of the training samples into the objective function for split optimization (see Equation (20)) as in Gall et al. (2011); Criminisi et al. (2011). Similarly, the sample prior is proportional to the vesselness likelihood L_V of Equation (17). The weight of the individual votes is thus decreased in regions of large uncertainty due to inaccuracies of the estimated vessel direction or image noise, for instance.

2.3.4. Centerline Extraction

In contrast to vessel classification, thresholding the centerline confidence map does not yield a topologically meaningful result, i.e., vessel centerline, but additional postprocessing is required. Moreover, one has to keep in mind that the regression framework accumulates displacement votes from surrounding samples within the vessel lumen only. The absolute value of the vote map hence depends on the local vessel diameter. Nonetheless, it can readily be used as an energy term to extract centerlines as minimal paths using a fast marching framework (Sethian, 1999). To this end, let $\mathbb{P}(\mathbf{p}, \mathbf{q})$ denote the set of naturally parametrized paths from \mathbf{p} to \mathbf{q} . The optimal centerline path w.r.t. the confidence map of Equation (24) can then be defined as

$$\hat{C}(\mathbf{p}, \mathbf{q}) = \arg \min_{P \in \mathbb{P}(\mathbf{p}, \mathbf{q})} \int_{t=0}^{|P|} \frac{1}{L_C(P(t))} dt \quad , \quad (26)$$

where $|P|$ is the arc length of path P . The strong formulation of this optimization problem leads to the Eikonal equation, which is a special case of the Hamilton–Jacobi equation (Cohen and Kimmel, 1997):

$$|\nabla U(\mathbf{x})| = \frac{1}{L_C(\mathbf{x})} \quad , \quad (27)$$

with boundary condition $U(\mathbf{p}) = 0$. This nonlinear hyperbolic partial differential equation models the (earliest) arrival time of a wave propagated from \mathbf{p} with

speed $L_C(\mathbf{x}) \geq 0$. A fast marching scheme based on upwind finite difference approximation as proposed by Sethian (1999) is used to efficiently solve the Eikonal equation. The minimal path between \mathbf{p} and \mathbf{q} is eventually extracted by back-propagation starting at \mathbf{q} until the global minimum at \mathbf{p} is reached (Sethian, 1999). We use fourth-order Runge–Kutta optimization rather than gradient descent steps on $U(\mathbf{x})$ to avoid oscillations (Benmansour et al., 2013).

3. Experiments

In this section, we describe the different experiments to evaluate the proposed framework for vessel segmentation and centerline extraction for real and synthetic data. If not otherwise stated, we will use the default parameters summarized in Table A.3. An exhaustive parameter study will be discussed in Section 4.

3.1. Vessel Segmentation

Datasets. We evaluate the performance of our segmentation framework on four 3-D datasets obtained from synchrotron radiation X-ray tomographic microscopy (SRXTM) of cylindrical samples of the rat somatosensory cortex (see Figure 1). The optical magnification was twentyfold resulting in a total volume size of $2048 \text{ px} \times 2048 \text{ px} \times 4000 \text{ px}$ with an isotropic voxel spacing of 700 nm for the reconstructed 16 bit grayscale images (Reichold et al., 2009). In a preprocessing step we apply anisotropic diffusion filtering in order to reduce image noise while preserving edge contrast (Perona and Malik, 1990). From each (preprocessed) dataset two disjoint regions of interest (ROIs) of size $256 \text{ px} \times 256 \text{ px} \times 256 \text{ px}$ are extracted for training and testing, respectively. In the following, we will refer to these non-overlapping ROIs as test and training data/ROI, respectively. For each test ROI, ground truth (GT) labels were manually generated by an expert assisted by a semi-automatic segmentation tool (Yushkevich et al., 2006) on 15 evenly distributed slices along each axis (axial, coronal, sagittal). Thus, a total of 180 2-D slices have been labeled containing 7.3×10^4 foreground and 2.7×10^6 background labels in average ($\pm 3.9 \times 10^4$) corresponding to a vascular volume fraction of $(2.7 \pm 1.5) \%$.

Baseline. In a first baseline experiment, all ROIs are segmented using Otsu’s method (Otsu, 1979) and two different multi-scale approaches based on vessel enhancement filtering (Sato et al., 1997; Frangi et al., 1998) and optimally oriented flux (OOF) for curvilinear structure detection as proposed by Law and Chung (2008). We perform an exhaustive grid search to optimize the vesselness scales on the test ROIs. To this end, the different experiments for varying scales were ranked w.r.t. the area under the receiver operating characteristic (ROC) curve using the GT labels of the test ROIs. The scale configuration yielding the minimum average rank index over the four datasets is ultimately chosen for the baseline experiments. For Frangi’s and Sato’s vesselness filters five logarithmically spaced scales performed best: $\sigma = (0.70, 1.1, 1.7, 2.7, 4.2) [\mu\text{m}]$. Following Law and Chung (2008), the OOF scale space was densely sampled according to the Nyquist sampling rate: $\sigma_k = (0.7 + 0.35 \cdot k) \mu\text{m} \forall_{0 \leq k \leq 26}$.

Feature Extraction. In a next step, we compute the OSF eigenfilters introduced in Section 2.1.1 from 3000 randomly sampled patches centered at voxels labeled as vessel in the Otsu label map. In particular, background patches are not considered for the estimation of the OSF templates. Besides the original vessel patches, five randomly rotated versions of each patch are added to the set of patches P_{OSF} used in Equation (1) in order to account for rotational symmetry of vessel structures – even in case of orientation bias in the training ROI – while keeping the total number of patches at a moderate level ($N_P = 1.8 \times 10^4$). As in (Menze et al., 2006), the OSF patch size P was assessed from the RF feature importance and set to $P = 19$ px. Similarly, rOSF templates are estimated from the same 3000 randomly sampled foreground patches. In contrast to OSF, each patch is transformed to the normalized pose as described in Section 2.1.2. To this end, the local vessel direction at each patch center is estimated based on the eigenanalysis of the Hessian computed at the most discriminative scale as defined by Frangi’s multi-scale vesselness (Frangi et al., 1998).

As for the SFT feature model, we perform a small parameter study to optimize the SFT scales similar to the multi-scale vesselness parameters. In order to avoid overfitting, however, we use the training ROIs for the parameter optimization where the Otsu labels are considered as ground truth in this case. We ultimately select $S = 3$ logarithmically spaced scales $\sigma = (0.70, 1.7, 4.2)$ [μm]. For Gaussian derivatives up to order $M = 1$ (2, 3, 4), the SFT model thus defines $S \cdot d_M = 9$ (27, 57, 102) features, respectively (see Equations (4) and (5)). For a fair comparison of the SFT and (r)OSF feature models, the PCA subspace dimension d , i.e., the number of (r)OSF features, has been chosen accordingly.

Training and Prediction. Different RF classifiers consisting of $N_T = 256$ decision trees were trained separately on the training ROI of a single dataset using OSF, rOSF, and SFT features along with orthogonal and oblique splits, respectively, as explained in Algorithm 1. The training was repeated for each dataset using $N_S = 4000$ foreground (vessel) and background samples, respectively, randomly drawn from the Otsu label map to define the sample set S . The local vessel direction, as required for RF training using rOSF and SFT features, is again estimated based on the Hessian eigenanalysis (see above). Note that we compute the (noisy) training labels fully automatically without any user input. The manually annotated GT labels for the test ROIs are used for validation only.

Finally, the different RF models were applied to the test ROIs of each dataset. The classification performance was then evaluated on the uniformly aligned slices with GT labels available (see above). The generalization error of the individual classifiers is investigated for “totally” unseen data using leave-one-out cross-validation on the four datasets, which will be referred to as “inter-dataset” validation in the following. Besides, we also perform “intra-dataset” validation by choosing the (non-overlapping) training and test ROI from the same dataset. In this way, we assess the prediction error on morphologically similar (yet still unseen) test data from the same dataset that has been used for RF training.

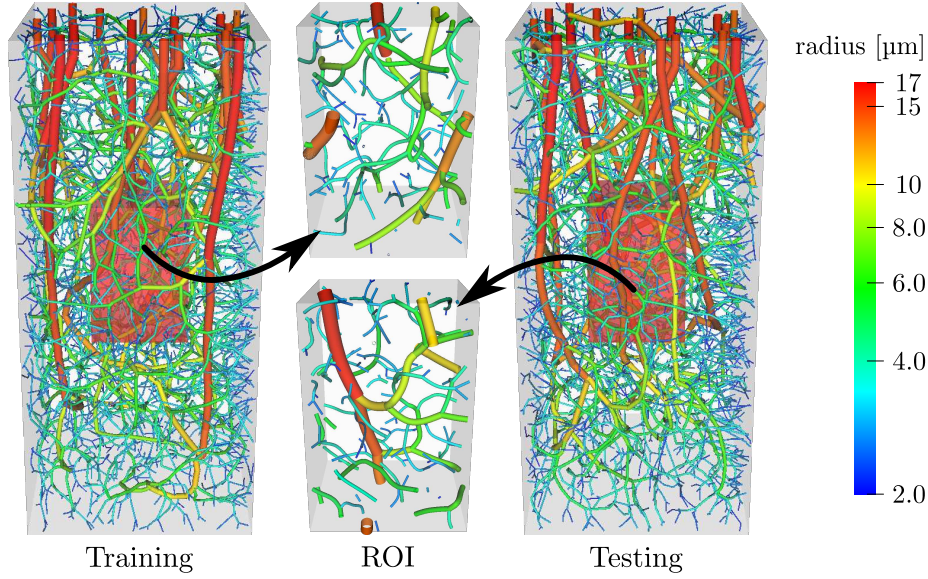


Figure 4: Synthetic arterial tree models used for training (left) and testing (right) (Schneider et al., 2012, cf. Fig. 5). The red cuboids in the center of the simulation domain (left, right) outline the ROI of the subgraphs used for our experiments (center). The vessel radii are color-coded on a logarithmic scale. The training and test ROI approximately cover the same range of vessel radii (train: $(2.0\text{--}15.6)\mu\text{m}$, test: $(2.0\text{--}16.1)\mu\text{m}$). [Best viewed in color]

3.2. Centerline Extraction

In a second set of experiments, we test the accuracy of our extended framework for centerline extraction using both synthetic and real datasets. The latter consists of the four datasets used for the segmentation experiments.

Datasets. Synthetic data is obtained as follows: An artificial, yet physiologically plausible, arterial tree model is generated for a box-shaped simulation domain as described in Schneider et al. (2012). The vascular network is represented as a discrete graph structure (rooted tree) where each vessel segment is modeled as rigid cylindrical tube with a constant radius inscribed in the vessel lumen. Spline smoothing (de Boor, 2001) is applied to the generated network before it is cropped to a centered cuboid with dimension $350\mu\text{m} \times 350\mu\text{m} \times 512\mu\text{m}$ (see Figure 4). The cropped network is considered as the ground truth vascular geometry for our experiments on synthetic data. Next, we synthesize 3-D image data from the vascular model. To this end, we first compute two binary label maps of size $387\text{px} \times 387\text{px} \times 565\text{px}$ with an isotropic voxel spacing of $1\mu\text{m}$ marking each voxel as (1) inside/outside the vessel lumen (segmentation map), and (2) part/not part of the discretized vessel centerlines (centerline map). Convolution of the segmentation map with a Gaussian point spread function ($\sigma = 1\mu\text{m}$, see Figure 5a) ultimately yields the synthetic 3-D image data. An axial slice is shown in Figure 5b with superimposed segmentation and centerline

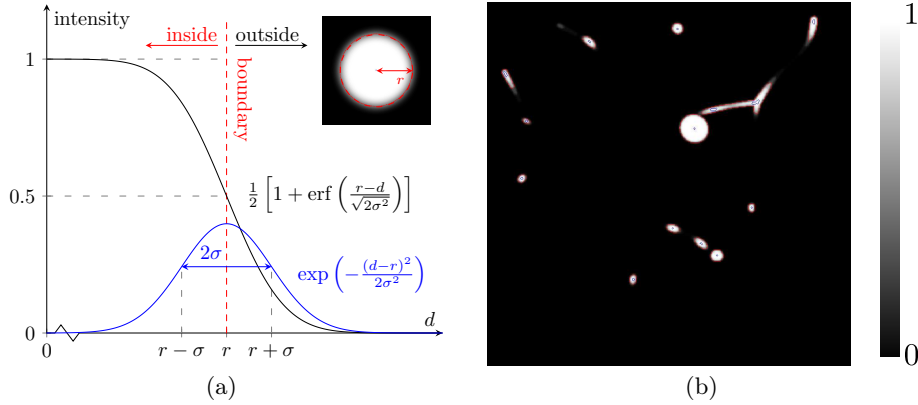


Figure 5: Synthesis of image data from artificial vascular model. (a) Intensity profile orthogonal to the vessel direction w.r.t. the distance d to the centerline for vessel radius r using a Gaussian point spread function. (b) Axial slice of the synthetic image data with the vessel boundaries and the rasterized centerline voxels superimposed in red and blue, respectively. [Best viewed in color]

labels. We generate two synthetic datasets for training and testing using different random seeds to drive the arterial tree simulation of Schneider et al. (2012). Thus, we obtain different yet morphologically similar training and test data (see Figure 4). We further verify that the radius histograms of the two synthetic datasets approximately cover the same range of scales and hence allow for a fair comparison with the OOF multi-scale approach where we densely sample the scale space (see below).

Training and Prediction. Similar to the classification experiments, we learn a Hough regression forest from 32,000 randomly sampled training exemplars that are labeled as vessel in the segmentation map (see Section 2.3). The corresponding centerline offsets are computed according to Equation (18) with the centerline C defined by the rasterized centerline map. SFT features are computed for Gaussian derivatives up to order $M = 4$ at three different scales $\sigma = (2.0, 4.4, 10)$ $[\mu\text{m}]$ with the local vessel direction estimated based on multi-scale Hessian eigenanalysis ($\sigma = (2.0, 3.1, 4.9, 7.7, 12)$ $[\mu\text{m}]$, see above). In contrast to RF classification, we increase the number of randomly chosen feature channels at each split node to $\lceil \frac{N_F^0}{3} \rceil$ as proposed in Liaw and Wiener (2002). For the 102 SFT feature channels in our experiment, this corresponds to 35 features per split (instead of 11 for classification).

We validate the regression framework for centerline extraction on the synthetic test dataset. Centerline votes are cast at all voxels labeled as foreground in the synthetic segmentation map with a uniform prior (see Equation (25)). The centerline segments are finally extracted from the voting map as minimal paths between the start/end node of all vessel segments as defined by the GT arterial tree model. Note that the GT geometry, i.e., location of seeds, and topology, i.e., connection of seeds, is usually not known for real data. The experiment on the

synthetic data hence focuses on the spatial accuracy that can be achieved based on the proposed voting framework rather than the topological correctness or seed point accuracy. As manual selection of the seed points is prohibitive for large datasets, we pursue a different strategy for the choice of the seed points for the real data (see below).

For the real datasets, we manually generate GT centerline labels. To this end, a topological skeleton is extracted from the binary segmentation obtained from the RF-SFT experiments described above (see Figure 11) using distance-ordered homotopic thinning (DOHT) (Palágyi and Kuba, 1998; Pudney, 1998). Briefly, the method iteratively peels off simple points within the surface layer of the object. A point is considered simple iff its removal does not alter the object topology. Following Pudney (1998), the object points are visited (and potentially deleted) in ascending order of their distance to the background computed by a distance transform of the inverted segmentation map. The candidate points are characterized as simple based on the binary pattern of their local neighborhood (Chen and Molloy, 2003). A discrete graph model is finally extracted from the skeletonized object. Skeletonization artifacts, e.g., spurious sprouts, are removed based on simple geometric heuristics. Based on the DOHT skeleton, the centerline is manually adjusted for selected vessel segments by an expert and will be used as ground truth in the experiments. Finally, each GT centerline point is assigned a radius estimate based on the distance map to the closest background voxel (see Table 2). The radius information is used to select proper scales for the baseline experiments (see below).

Training and testing of the RF classifiers for the real datasets proceeds similar to the experiments on synthetic data. Training exemplars are sampled from the RF-SFT segmentation map, centerline offsets are computed from the (automatically extracted) DOHT skeleton. The SFT feature scales are the same as for the segmentation experiment ($\sigma = (0.7, 1.7, 4.2)$ [μm]). Note that the features have to be computed only once for segmentation and centerline extraction. During prediction, the RF-SFT vessel confidence map from the previous segmentation experiments is used as prior in Equation (25). We extract optimal centerline segments w.r.t. Equation (26), where the seeds are defined by the vessel segments of the DOHT skeleton. Thus, each vessel segment of the DOHT skeleton and the corresponding extracted centerline share their start and terminal node. We perform a leave-out-one cross-validation test using three datasets for training and the remaining for testing.

Baseline. As baseline experiments, we choose two approaches based on fast marching as well, but using different speed functions. In the first approach the speed function is defined based on the distance to the background (DIST). By design, the propagated waves between the start and end point of a centerline candidate hence travel faster in the vessel center (Deschamps, 2001). The second approach is a simplified version of Benmansour and Cohen (2011) tracing geodesic paths in the scale-space domain (Benmansour et al., 2013). Briefly, a tubularity measure based on OOF (Law and Chung, 2008) is computed at multiple scales densely sampling the scale space in the range of the tubular structures to be

detected. The OOF value at a specific location and scale can be interpreted as the likelihood of the voxel being centered in a tubular structure of the selected scale. For the synthetic datasets, we compute the tubularity map for 15 scales $\sigma = (2.0, 3.0, \dots, 16)$ [μm]. Note that the vessel radii of the test ROI cover the same range (see Figure 4). Likewise, 12 scales $\sigma = (0.7, 1.4, \dots, 8.4)$ [μm] are used for the real datasets covering the radius range of all test sets (see Table 2).

The accuracy of the extracted centerlines in all experiments is assessed based on an absolute and relative distance metric:

$$\begin{aligned} D_{\text{abs}}(C, C^0) &= \frac{1}{|C|} \sum_{\mathbf{p} \in C} \min_{\mathbf{q} \in C^0} \|\mathbf{p} - \mathbf{q}\| \\ D_{\text{rel}}(C, C^0) &= \frac{1}{|C|} \sum_{\mathbf{p} \in C} \frac{1}{r_{\text{NN}}} \min_{\mathbf{q} \in C^0} \|\mathbf{p} - \mathbf{q}\| \quad , \end{aligned} \quad (28)$$

where C and C^0 denote, respectively, the extracted and reference centerline graphs and r_{NN} the radius at the nearest neighbor in C^0 .

4. Results and Discussion

In this section we present and discuss the experimental results. First, we analyze the orthogonal subspace filters learned from our data and compare them to steerable filters in Section 4.1. Section 4.2 discusses the training of classification trees for both types of features and presents qualitative and quantitative segmentation results for the presented framework and the baseline experiments. In Section 4.3 we explore the parametrization of classification forests in an exhaustive parameter study. Similarly, we present qualitative and quantitative results for the experiments on centerline extraction (Section 4.4) along with a sensitivity analysis of the parametrization of Hough regression forests (Section 4.5).

4.1. PCA-Based Matched Filters

When inspecting the OSF filter templates we observe a highly structured pattern (see Figure 6a). The ball-shaped mean shows a Gaussian-like pattern. The most significant principal axis captures the average image intensity in the vicinity of the sample. Broadly speaking, patches $\alpha_2, \dots, \alpha_4$ capture first-order derivatives along the anterior-posterior (A-P), right-left (R-L), and superior-inferior (S-I) direction, respectively. Similar first-order patterns at smaller scales and along different directions appear in $\alpha_{10}, \dots, \alpha_{14}$. Patches $\alpha_5, \dots, \alpha_9$ correspond to differently oriented second-order derivatives.

The rOSF templates of Figure 6b reveal an even richer structure of the data at hand. The mean patch captures the Gaussian intensity profile orthogonal to the normalized vessel direction (R-L). The multi-scale nature of the analyzed vascular structure is reflected in the most significant principal axis α_1 and α_5 . Similar to OSF, first-order derivatives at different scales and orientations are captured in $\alpha_2, \alpha_3, \alpha_4$, and α_6 . Note however, that the derivatives are mostly aligned

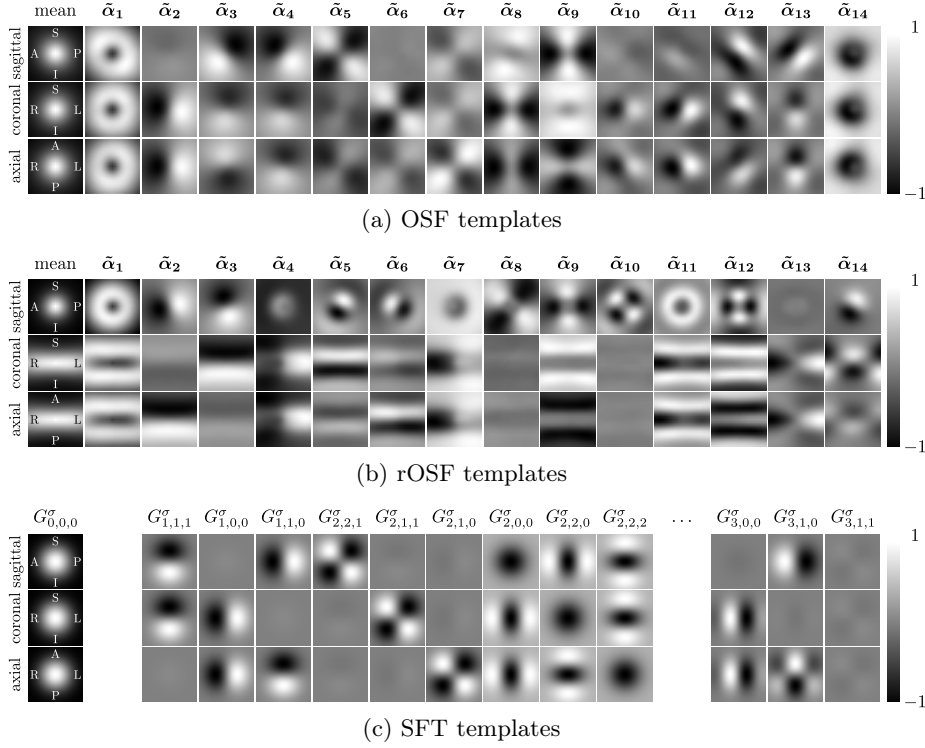


Figure 6: Filter templates of different feature models visualized for centered sagittal, coronal, and axial slices. (a,b) Mean pattern and most significant (reshaped) eigenfilters $\tilde{\alpha}_{\mathbf{k}}$ computed from vessel patches before (a) and after (b) pose normalization (R-L direction). (c) SFT filter templates defined as normalized Gaussian derivatives $G^{\sigma}_{m,a,b}$ of different orders at a single scale σ (see Equation (3)). The parametrized SFT templates feature similar structural properties as the learned eigenspace filters.

along the A-P and S-I direction, while derivatives along the vessel direction (R-L) appear later (α_2 , α_{13}). Likewise, second- and third-order derivatives orthogonal to the vessel direction are captured as well. For comparison, normalized Gaussian derivatives of different orders as used for the SFT feature model are visualized in Figure 6c. They feature similar structural properties as OSF templates with the additional advantage of explicit multi-scale parametrization and steerability.

The PCA spectra show a sharp profile as indicated in Figure 7a. Hence, the variance of the vessel patches can be described by few modes only. It also becomes obvious that the spectrum drops significantly faster for patches explicitly aligned w.r.t. the local vessel direction (rOSF). In this way, the intrinsic structure of the rOSF patches is reduced by one degree of freedom (orientation) which does no longer need to be captured by the PCA model as in the OSF case.

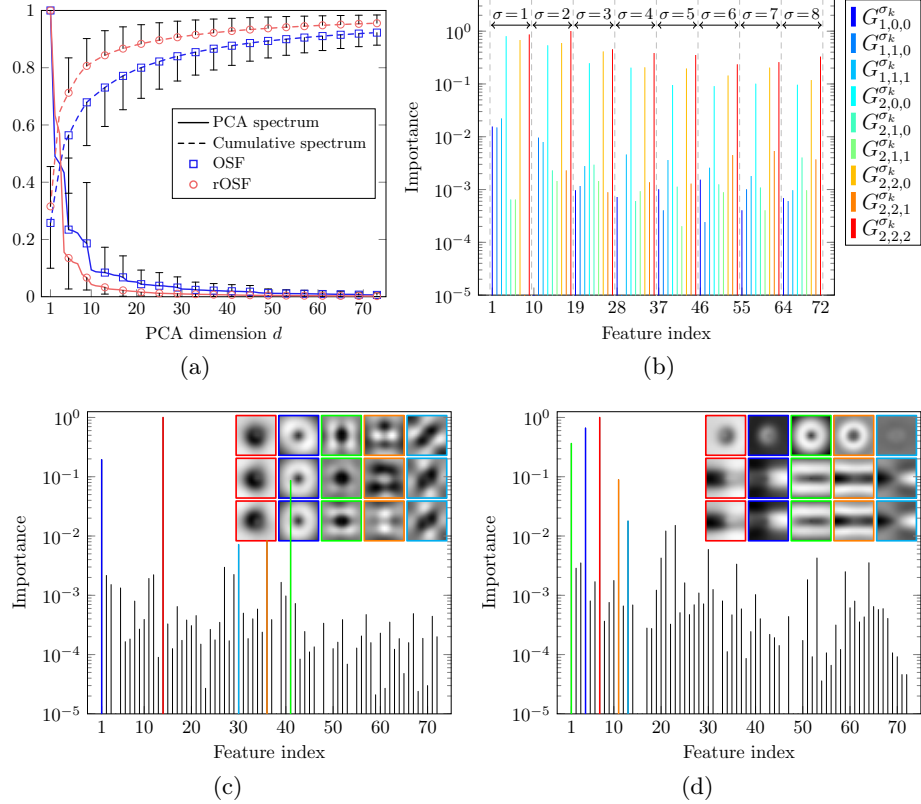


Figure 7: (a) Normalized PCA spectrum λ_k/λ_1 and explained variance as measured by the cumulative spectrum $\sum_{k=1}^d \lambda_k / \sum_{k=1}^{P^3} \lambda_k$ for (r)OSF patches, where λ_k denotes the k -th eigenvalue of the data covariance matrix. The values are averaged over all four datasets. Error bars indicate the standard deviation. Note that the error bars are plotted in one direction only for better visibility. (b-d) Variable importance (Breiman, 2001) for different feature models on a logarithmic scale (oblique splits): (c) OSF and (d) rOSF feature model using the most significant $d = 72$ PCA templates. Note that the feature indices correspond to the numbering in (a), i.e., features are sorted in descending order of the corresponding eigenvalue. Similar to Figure 6, the filter patches corresponding to the five feature channels with maximum importance (highlighted in color) are plotted in the upper right corner in the order of decreasing importance. (b) Importance of SFT features at eight different scales $\sigma = (1, \dots, 8)$ [px] up to order $M = 2$ (total of 72 features). The prominent peaks in (b) correspond to the Gaussian derivatives $G_{2,0,0}^\sigma$, $G_{2,2,0}^\sigma$, and $G_{2,2,2}^\sigma$. [Best viewed in color]

4.2. Vessel Segmentation

Feature Importance. We use the normalized RF feature relevance score, i.e., the permutation importance as defined in Breiman (2001), to identify the relevant scales and orders. It is a measure for the predictive power of the different features and is thus often used for feature selection. Figures 7b–d show the normalized importance scores for the different feature models using oblique splits. The SFT feature importance indicates that the second-order derivatives parallel and orthogonal to the vessel direction ($G_{2,0,0}^\sigma$, $G_{2,2,0}^\sigma$, $G_{2,2,2}^\sigma$) are most significant for the classification. Note that the Hessian-based segmentation approaches also rely on these features (Sato et al., 1997; Frangi et al., 1998). For increasing scales σ , the importance values tend to decline in general. In contrast to the PCA spectra, the (r)OSF feature importance shows a rather noisy profile. This clearly indicates that the discriminative power of a PCA mode, i.e., feature importance, is not necessarily dependent on the preserved variance of the data, i.e., feature index. Nonetheless, features ranking high w.r.t. the feature importance tend to be also more important w.r.t. the explained variance (see Figures 7c and d). With increasing feature index, the importance slowly decreases, in general, as the corresponding filter patches slowly start to model noisy structures. For both feature models, (r)OSF patches capturing the average image intensity in the local neighborhood are among the most discriminative features as a result of the multi-scale nature of the data at hand. Likewise, first-order derivatives as well as higher order structures seem to be highly discriminative. Interestingly, first-order derivatives along and orthogonal to the local vessel direction (rOSF) seem to be more important than second-order structures as in the SFT case.

Segmentation Results. Comparing the overall classification performance of the proposed learning-based approaches for different model parameters to standard segmentation approaches reveals the superior performance of the SFT features as indicated by the precision-recall curves (PRCs) in Figure 8. With regard to the different feature models, the PRCs confirm that SFT features clearly outperform the (r)OSF model. It is remarkable that rOSF features perform worse compared to the OSF model, in general. Only for a rather low-dimensional feature space ($d = 9$), rOSF yields better results. Increasing the number of features does not improve the performance significantly as indicated by the narrow area in Figure 8 shaded in dark gray. We will discuss this observation in further detail later. In addition, the plot shows that the RF-SFT model is superior to the OOF and Hessian-based approaches, even for a small number of features. Note that the reported results for OOF, Frangi, and Sato have to be considered as upper bound as the scale parameters have been optimized on the test data (overfitting). The analysis also reveals that the segmentation performance of the RF-SFT model hardly changes if we choose the maximum derivative order of the SFT features $M > 1$ (see Figure 8, center), which is consistent with the observation of the second-order derivatives being the most discriminative features (see Figure 7b). For some cases, the RF-SFT model for $M = 3$ even performs worse compared to $M = 2$. Assessing the feature importance (see Figure 9) reveals that stepping from $M = 2$ to $M = 3$ mostly adds unimportant features

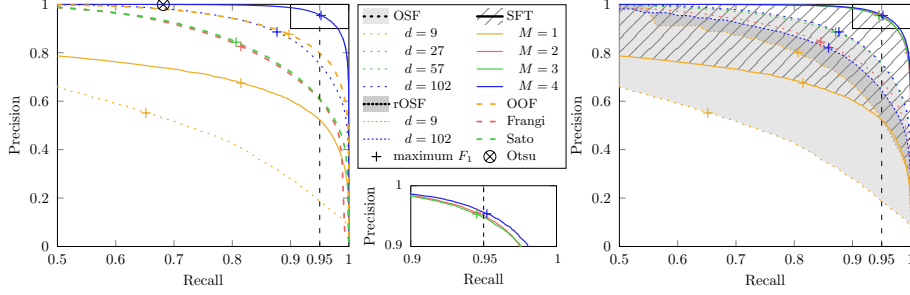


Figure 8: Precision-recall curves (PRCs) and optimal operating points w.r.t. F_1 measure for a single dataset. Left: RF-OSF and RF-SFT in comparison to optimally oriented flux (OOF), Frangi’s/Sato’s vesselness, and Otsu thresholding. Right: Comparison of different feature models for varying number of features. The PCA dimension d is chosen to match the number of SFT features with Gaussian derivatives up to order M at three different scales (see text). The differently shaded areas outline the boundaries between the worst- and best-performing PRC of the three feature models. Center: Scaled version of the upper right corners of the main plots (black rectangle). [Best viewed in color]

(see Figure 9b) which can make the randomized feature selection less effective. Incorporating derivatives of fourth-order, however, introduces additional features that turn out to be more significant, most importantly fourth and mixed second-order derivatives along and orthogonal to the vessel direction, e.g., $G_{4,0,0}^{\sigma}$ and $G_{4,2,2}^{\sigma}$. In spite of the significantly increased number of features for $M = 4$ (102 features) in comparison to $M = 2$ (27 features), the segmentation performance hardly changes, which once more emphasizes the importance of the second-order derivative features.

A more detailed numerical analysis of the classification performance of the different approaches is summarized in Table 1 and confirms the superior performance of the RF-SFT model over the (r)OSF features and the multi-scale OOF and vesselness approaches. Otsu’s method (Otsu, 1979) tends to underestimate the global threshold, which results in a conservative segmentation with high precision (PPV), maximum specificity (SPC), and very few false positives (FPR) only. However, the segmentation of vessel boundaries, in particular, becomes inaccurate as indicated by the increased balanced error rate (BER) and the F_1 measure. The average segmentation performance on “totally” unseen data (inter-dataset) consistently decreases as compared to the intra-dataset analysis using morphologically similar test and training data from disjoint ROIs of the same dataset (see Section 3.1). In the case of the RF-SFT model, in particular, these differences become negligible, which shows that the RF classifier generalizes well. The figures also reveal that oblique splits, as compared to orthogonal splits, yield both better classification performance and smaller (average) path lengths regardless of the chosen feature model. The advantage of oblique over orthogonal splits may result from the correlation between the features, and correlated noise that is better captured by the multivariate oblique node model than by univariate splits (Menze et al., 2011). We define the average

Method	Validation	PPV [%]	SPC [%]	FPR [%]	BER [%]	F1 [$\times 10^{-2}$]	AUC-PR [$\times 10^{-2}$]	Dice	Jaccard	Path Length
RF-OSF	inter-data	70.68 \pm 8.70	98.94 \pm 0.66	1.06 \pm 0.66	3.08 \pm 0.34	80.75 \pm 5.69	44.31 \pm 2.09	0.81 \pm 0.06	0.68 \pm 0.08	7.63 \pm 0.63
	intra-data	75.53 \pm 4.60	99.23 \pm 0.22	0.77 \pm 0.22	2.93 \pm 0.12	84.06 \pm 2.86	45.60 \pm 1.22	0.84 \pm 0.03	0.73 \pm 0.04	
RF-rOSF	inter-data	65.87 \pm 7.29	98.63 \pm 0.88	1.37 \pm 0.88	3.22 \pm 0.45	77.56 \pm 5.17	42.33 \pm 2.90	0.78 \pm 0.05	0.64 \pm 0.07	7.31 \pm 0.48
	intra-data	68.59 \pm 9.02	98.94 \pm 0.18	1.06 \pm 0.18	3.06 \pm 0.10	79.40 \pm 5.93	43.32 \pm 2.54	0.79 \pm 0.06	0.66 \pm 0.08	
RF-SFT	inter-data	91.05 \pm 1.72	99.76 \pm 0.11	0.24 \pm 0.11	2.67 \pm 0.06	92.94 \pm 0.91	48.62 \pm 0.28	0.93 \pm 0.01	0.87 \pm 0.02	5.98 \pm 0.91
	intra-data	92.25 \pm 1.17	99.78 \pm 0.12	0.22 \pm 0.12	2.65 \pm 0.06	93.56 \pm 0.62	48.81 \pm 0.17	0.94 \pm 0.01	0.88 \pm 0.01	
RF-OSF	inter-data	77.03 \pm 5.86	99.28 \pm 0.26	0.72 \pm 0.26	2.89 \pm 0.14	84.94 \pm 3.59	45.86 \pm 1.28	0.85 \pm 0.04	0.74 \pm 0.05	4.61 \pm 0.32
	intra-data	79.52 \pm 2.91	99.37 \pm 0.25	0.63 \pm 0.25	2.85 \pm 0.13	86.52 \pm 1.74	46.61 \pm 0.57	0.87 \pm 0.02	0.76 \pm 0.03	
RF-rOSF	inter-data	69.15 \pm 4.56	98.89 \pm 0.47	1.11 \pm 0.47	3.08 \pm 0.23	79.94 \pm 3.02	43.38 \pm 1.68	0.80 \pm 0.03	0.67 \pm 0.04	4.37 \pm 0.21
	intra-data	70.68 \pm 7.38	99.04 \pm 0.16	0.96 \pm 0.16	3.00 \pm 0.08	80.88 \pm 4.72	43.94 \pm 1.97	0.81 \pm 0.05	0.68 \pm 0.07	
RF-SFT	inter-data	95.35 \pm 1.73	99.88 \pm 0.04	0.12 \pm 0.04	2.60 \pm 0.04	95.13 \pm 0.86	49.21 \pm 0.24	0.95 \pm 0.01	0.91 \pm 0.02	4.15 \pm 0.78
	intra-data	95.91 \pm 1.22	99.89 \pm 0.06	0.11 \pm 0.06	2.58 \pm 0.04	95.42 \pm 0.59	49.31 \pm 0.15	0.95 \pm 0.01	0.91 \pm 0.01	
OOF	average	79.27 \pm 1.99	99.29 \pm 0.49	0.71 \pm 0.49	2.86 \pm 0.25	86.42 \pm 1.19	46.35 \pm 0.27	0.86 \pm 0.01	0.76 \pm 0.02	n/a
Sato	average	63.50 \pm 2.65	98.54 \pm 0.72	1.46 \pm 0.72	3.23 \pm 0.36	76.10 \pm 1.91	42.63 \pm 0.68	0.76 \pm 0.02	0.61 \pm 0.02	n/a
Frangi	average	60.71 \pm 2.05	98.32 \pm 0.95	1.68 \pm 0.95	3.34 \pm 0.47	74.06 \pm 1.52	41.75 \pm 0.54	0.74 \pm 0.02	0.59 \pm 0.02	n/a
Otsu	average	99.96 \pm 0.03	100.00 \pm 0.00	0.00 \pm 0.00	14.29 \pm 1.57	83.29 \pm 2.13	n/a	0.83 \pm 0.02	0.71 \pm 0.03	n/a
PPV: positive predictive value (precision), SPC: specificity, FPR: false positive rate, BER: balanced error rate, AUC-PR: area under the precision-recall curve										

Table 1: Quantitative evaluation of the segmentation performance for different approaches using “intra-dataset” and “inter-dataset” cross-validation (see text). The reported numbers are averaged over all datasets (mean \pm standard deviation). The operating point of the RF classifiers has been selected at the 95 % recall level (see Figure 8). The partial area under the precision-recall curve (AUC-PR) has been computed on the recall interval [0.5, 1]. The balanced error rate (BER) is reported as defined by Chen and Lin (2006). Figures highlighted in bold face mark the best result w.r.t. the corresponding metric. The oblique split model outperforms univariate orthogonal splits w.r.t. segmentation performance and average path length. Likewise, SFT features prove to be superior to (r)OSF. Otsu’s conservative segmentation achieves high precision (PPV), maximum specificity (SPC), and very few false positives (FPR) at the expense of an inaccurate segmentation of vessel boundaries (false negatives) as indicated by the increased balanced error rate and the F_1 measure.

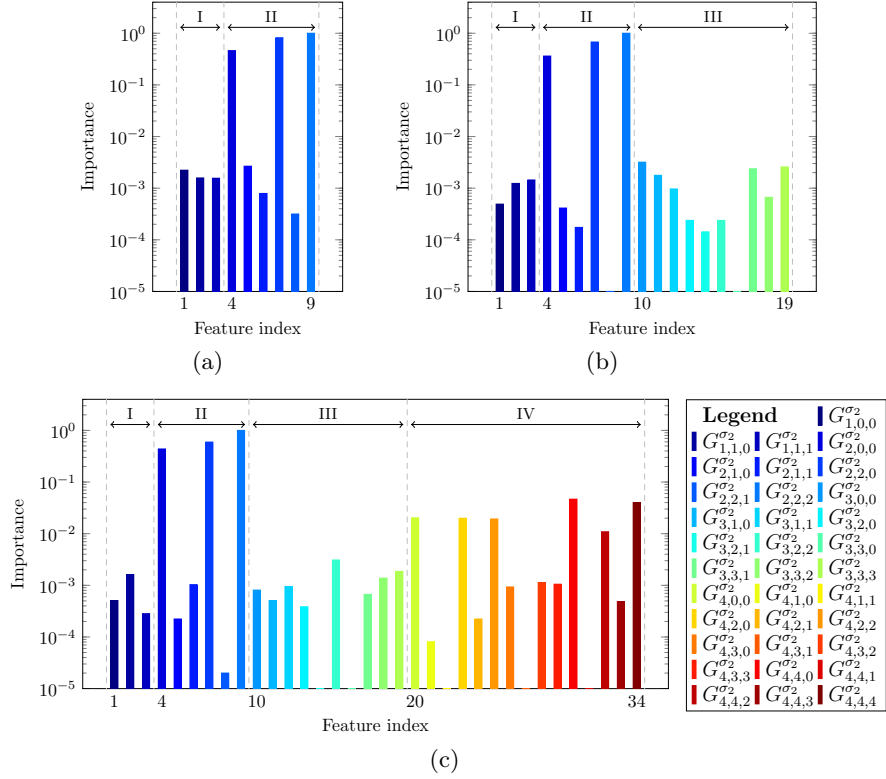


Figure 9: Variable importance (Breiman, 2001) of the RF-SFT model on a logarithmic scale for the intermediate scale $\sigma_2 = 1.7 \mu\text{m}$ and different maximum derivative orders (a) $M = 2$, (b) $M = 3$, (c) $M = 4$. The feature ranges labeled by roman numbers I–IV correspond to the Gaussian derivatives of the respective order. Second-order derivatives are most important in all test. [Best viewed in color]

forest path length (FPL) as the weighted tree depth over all leaf nodes averaged over all trees of the RF classifier:

$$\text{FPL} = \frac{1}{N_T} \sum_{t=1}^{N_T} \frac{1}{\sum_{l \in L_t} |\mathcal{T}_l|} \sum_{l \in L_t} |\mathcal{T}_l| \cdot d_l \quad , \quad (29)$$

where L_k denotes the k -th tree, \mathcal{T}_l and d_l the training examples and depth at leaf node l , respectively. Thus, FPL is a measure for the expected number of splits required to push a random exemplar from the root to the leaf node.

In order to gain further insight into the complex, deeply nested tree structure, we assess the pairwise sample proximity. More specifically, we accumulate a proximity matrix for 128 randomly sampled test exemplars per class, i.e., matrix size 256×256 . For each of the N_T decision trees, any pair of test examples sharing a leaf node has their proximity increased by $1/N_T$. Figure 10a shows the proximity matrices for the orthogonal and oblique node model using RF-SFT

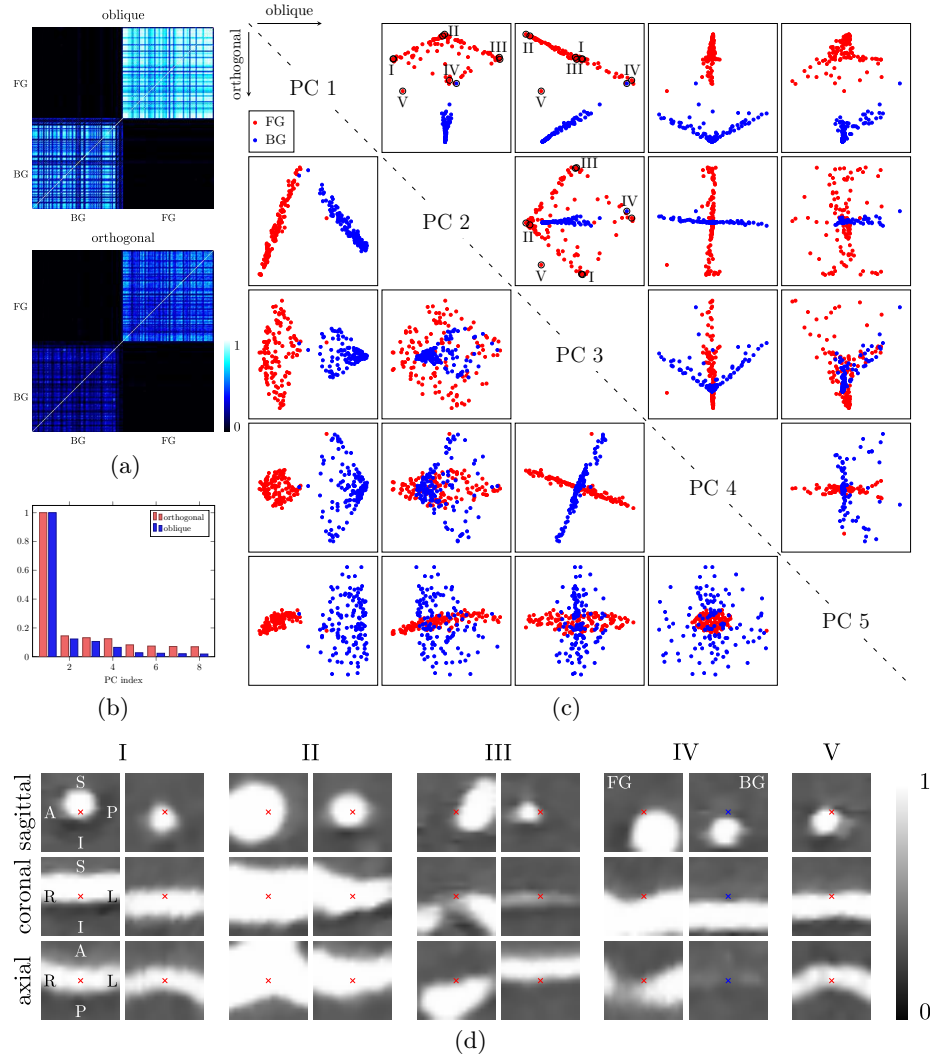


Figure 10: Sample proximity for the oblique and orthogonal node model using RF-SFT with default parameters. The proximities are computed for 128 randomly sampled test examples per class (FG: foreground, BG: background). (a) Co-occurrence matrices specifying the relative frequency of two test samples sharing the same terminal node. (c) Proximity plot w.r.t. five most significant principal components (PCs) computed from multidimensional scaling (MDS) of the proximity matrix. (b) Normalized spectrum of MDS modes. (d) Local image patches for selected test samples (black circles) from different regions (roman numbers) marked in (c). The patches are visualized for centered sagittal, coronal, and axial slices w.r.t. the normalized vessel direction (R-L). The patch centers are marked by small crosses. Oblique RF proximity explains variation (in MDS embedding) with fewer eigenspaces. [Best viewed in color]

with default parameters. The proximity matrix can be considered as a distance measure between test samples from an RF point of view. For better visualization, the matrix is then represented w.r.t. the most significant eigenmodes using multidimensional scaling (MDS) (Hastie et al., 2009). The proximity plots of Figure 10c for the different node models show that the test samples almost perfectly split along the first principal component. The second mode of the oblique split model captures additional structure in the foreground samples while the meaning of the corresponding mode in the orthogonal split model remains unclear. Comparing the spectra of the eigenmodes for the different split models in Figure 10b clearly indicates that the oblique RF proximity explains variations in the MDS embedding with fewer eigenspaces. Figure 10d illustrates the local image patches for selected test samples from different regions of the proximity plot in Figure 10c. The second and third eigenmode essentially capture the location of the foreground sample relative to the vessel center. Samples in region II appear pretty much centered in the vessel lumen, whereas the remaining regions (I,III,IV,V) correspond to samples close the surface.

In Figure 11 we compare the binary segmentation of the cerebrovascular network of a single dataset obtained by the different approaches using the F_1 -optimal operating points marked in Figure 8. Visually, Otsu’s method is too conservative while the Frangi filter and partly also the RF-OSF model generate rather smooth vessel surfaces missing some of the details. The ideal elliptical appearance model underlying the Hessian-based vesselness filters generate many false negatives at bifurcations, in particular, where the model assumptions do not hold. Here, the classification approach is able to consider more complex geometries, that are in accordance with higher order filter responses in the training data. As already indicated by the precision-recall analysis, the OOF-based segmentation proposed by Law and Chung (2008) outperforms the vesselness approaches but still produces severe “leakage” artifacts. In contrast, the RF-SFT results shown in the axial views are in much better agreement to the GT labels. Comparing OSF and rOSF features, the RF-rOSF confidence map tends to be much sharper at the vessel boundaries and partly reduces the noisy response of the OSF maps (region C). However, the rOSF segmentation is affected by severe leakage artifacts in some cases, e.g., top of region C, which results in over-segmentation. This is also reflected in the increased false positive rate (FPR) of rOSF over OSF in Table 1. The leakage might result from missing structural information in the training data or imperfect estimates of the local vessel direction. The latter has a direct impact on the computation of the local rOSF descriptor whereas the OSF features intrinsically model the anisotropy.

Computational Complexity. As described in Section 2.1, SFT has several advantages over (r)OSF including steerability and, more importantly, separability, which allows for efficient feature extraction. Comparing effective timings for the computation of the different features in Figures 12a and b clearly confirms the computational advantage of SFT over (r)OSF. The lack of separability of the (r)OSF filters results in a cubic computational complexity in the patch size ($\mathcal{O}(P^3)$) while SFT features can be extracted in linear time $\mathcal{O}(3P)$ (see

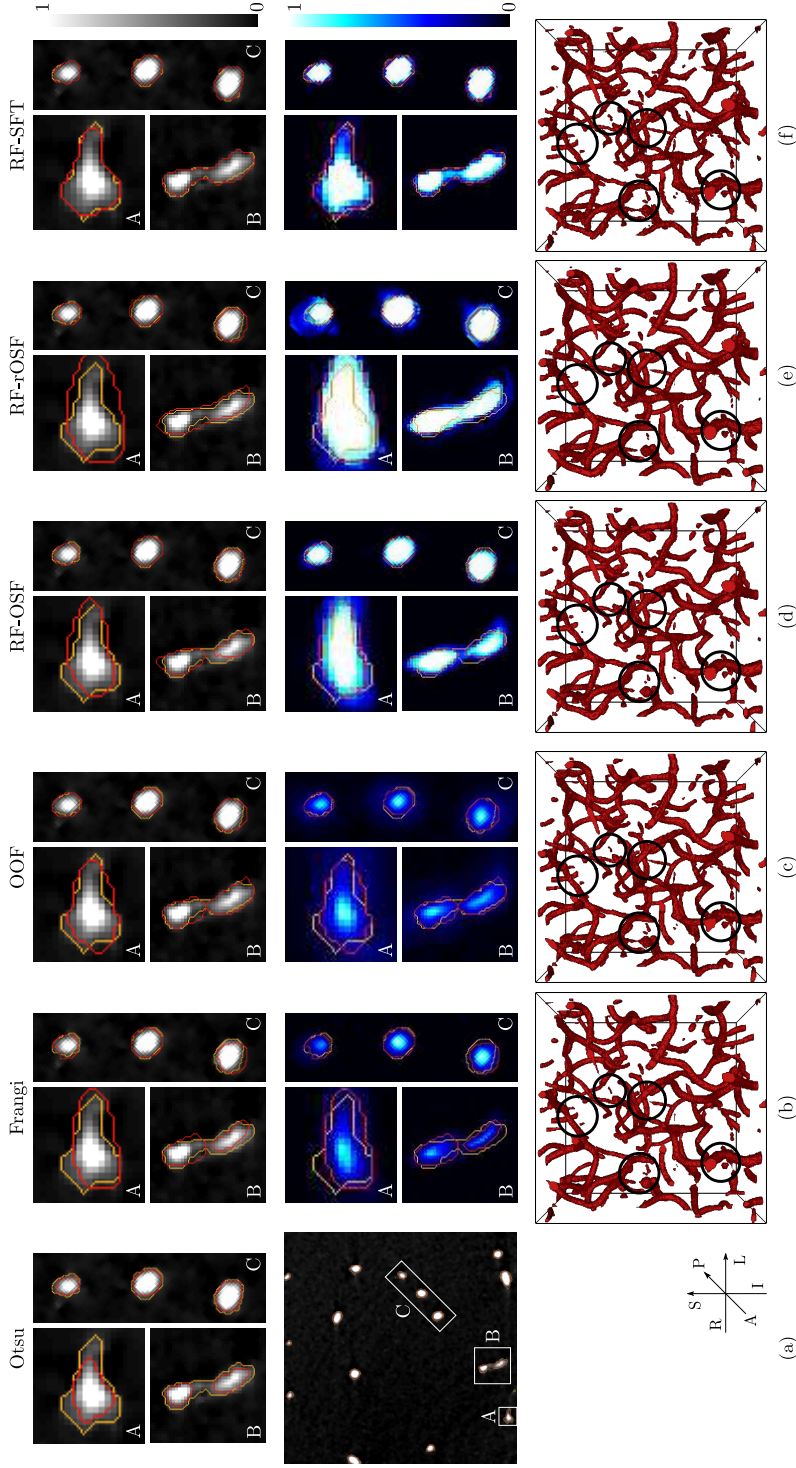


Figure 11: Visualization of the segmented cerebrovascular network for the 3-D test ROI (bottom) and a single axial slice (top/middle) using different segmentation techniques (columns). (a) Global thresholding using Otsu's method (Otsu, 1979). (b) Hessian-based segmentation using Frangi's vesselness (Frangi et al., 1998). (c) Optimally oriented flux (OOF) proposed by Law and Chung (2008). (d,f) RF-based segmentation for different features: (d) OSF features considering the first $d = 102$ most significant filter templates. (e) rOSF features computed in normalized vessel orientation ($d = 102$). (f) SFT features using Gaussian derivatives up to order $M = 4$ at three different scales (see text). The bottom row shows (b,c) the corresponding F_1 -optimal operating points marked in Figure 8. The results are rendered in 3-D (bottom) and outlined in red (top/middle) along with the ground truth (GT) contours in orange for three subregions within the axial slice (A-C). Black circles in the 3-D plots highlight prominent differences in the segmentation results. [Best viewed in color]

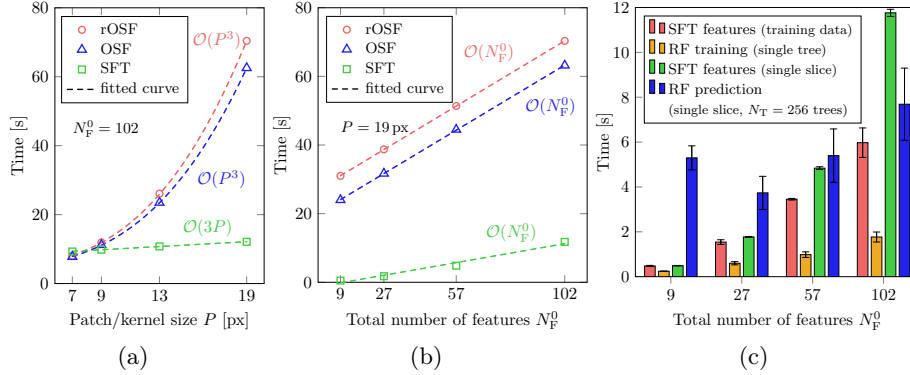


Figure 12: Computation time for feature extraction on a single 2-D slice of the real dataset ($256 \text{ px} \times 256 \text{ px} \times 256 \text{ px}$) for varying patch ((r)OSF) and Gaussian kernel (SFT) size P (a) and different number of features N_F^0 (b). The reported time measurements are averaged over all 256 slices. The dashed lines represent linear and cubic polynomials fitted to the data, respectively. (c) Computation time for training and prediction based on the RF-SFT model with default parameters (see Table A.3) averaged over all four test ROIs, standard deviation indicated by error bars. Timings are obtained from a standard desktop computer (CPU at 2.6 GHz with 8 GB of RAM) using our non-optimized C++ implementation based on ITK (Johnson et al., 2013) with multi-threading disabled for better comparison.

Figure 12a). For constant patch size $P = 19 \text{ px}$, as in the segmentation experiments, feature extraction of the different features is linear in the number of features ($\mathcal{O}(N_F^0)$) where SFT shows a smaller constant factor as compared to (r)OSF (see Figure 12b). Due to the lack of steerability, the computation of rOSF features requires local image resampling with regard to the estimated local vessel direction, which significantly adds to the total computational complexity of rOSF. The computation time to learn and apply the RF-SFT model are summarized in Figure 12c. As the decision trees can be trained simultaneously for a given training set, the RF training times are reported for a single tree. The timings for feature extraction, training and testing increase with the number of features, in general. As the prediction time also critically depends on the path length (number of splits) of the test samples from the root to the leaf node of the decision trees, the test time slightly decreases when stepping from $N_F^0 = 9$ to $N_F^0 = 27$. For $N_F^0 = 9$, i.e., maximum Gaussian derivative $M = 1$, the learned RF models have much deeper trees as the highly discriminative features computed from second-order derivatives ($N_F^0 = 27$, $M = 2$) are not considered. The total processing time for the segmentation of a volume of size $256 \text{ px} \times 256 \text{ px} \times 256 \text{ px}$ based on the proposed RF-SFT model is in the order of half an hour on a standard desktop computer. Segmentation of huge datasets ($\approx 10 \times 10^{10} \text{ px}$) requires about (12–18) h on a small cluster with 16 nodes. The large variation of the total wall-clock processing time can mostly be attributed to the computational overhead for data handling and input/output operations that can result in substantial idle time.

4.3. Parametrization of the Classification Forest

The proposed segmentation framework involves few user-defined parameters only. In contrast to other classifiers, such as the widely used support vector machine, e.g., González et al. (2009b); Rigamonti et al. (2011), the parameters related to the RF classifier, in particular, have an intuitive meaning and can easily be selected or optimized in a reasonable range. The sensitivity of the segmentation results with regard to the most important RF parameters has been analyzed in several parameter sensitivity analysis. To this end, we repeat the validation experiment for the RF-SFT model on the four datasets as described above using the default parameter configuration (see Table A.3) for all but one RF parameter which is varied in a reasonable range for each experiment. The resulting segmentation performance w.r.t. the (partial) area under the precision-recall curve (AUC-PR) is shown in Figures 13 and 14 for the different parameters. As in Table 1, the average AUC-PR is reported for inter- and intra-data cross-validation. In all parameter studies, the inter-data performance is slightly worse compared to the intra-data validation but shows very similar sensitivity characteristics.

Ensemble Size. As already noted by Breiman (2001) in his original paper, RF does not overfit as more trees are added to the ensemble. Figure 13a confirms that for an increasing *number of decision trees* (N_T), the average segmentation performance increases and tends to “converge” while the variance decreases. In our application, we found $N_T = 256$ to perform reasonably well while further increasing the number of trees did not result in remarkable improvements. Also note that the RF training (and testing) can easily be parallelized. The choice of N_T hence does not significantly affect the overall computational time.

Subspace Dimensionality. The *number of (random) feature channels* (N_F) that are individually selected at each split node to estimate the optimal split function (see Equation (15)) can be crucial for the performance of the RF classifier (see Figure 13b). Choosing N_F too small can make it difficult to robustly estimate the “optimal” splits as the (few) selected feature channels might not be discriminative. Repeated split optimization for different randomly chosen features as described in Algorithm 1, however, is able to partly overcome this problem. On the other hand, employing too many features at a time can make the split estimation unstable and lead to overfitting. We clearly see from Figure 13b that the widely used default choice $N_F = \lceil \sqrt{N_F^0} \rceil$ with N_F^0 being the total number of features (Liaw and Wiener, 2002), in fact yields a good overall performance and is close to the maximum.

Tree Structure. Commonly, decision trees are fully grown during training without pruning (Breiman, 2001). When dealing with very noisy feature or label data, however, it might be beneficial to limit the *maximum tree depth* (d_{\max}) as a regularizer. Figure 13c shows that pruning deteriorates the segmentation performance in our case. It increases with d_{\max} until convergence at about $d_{\max} = 8$ as the fully grown decision trees are rather flat as indicated by the

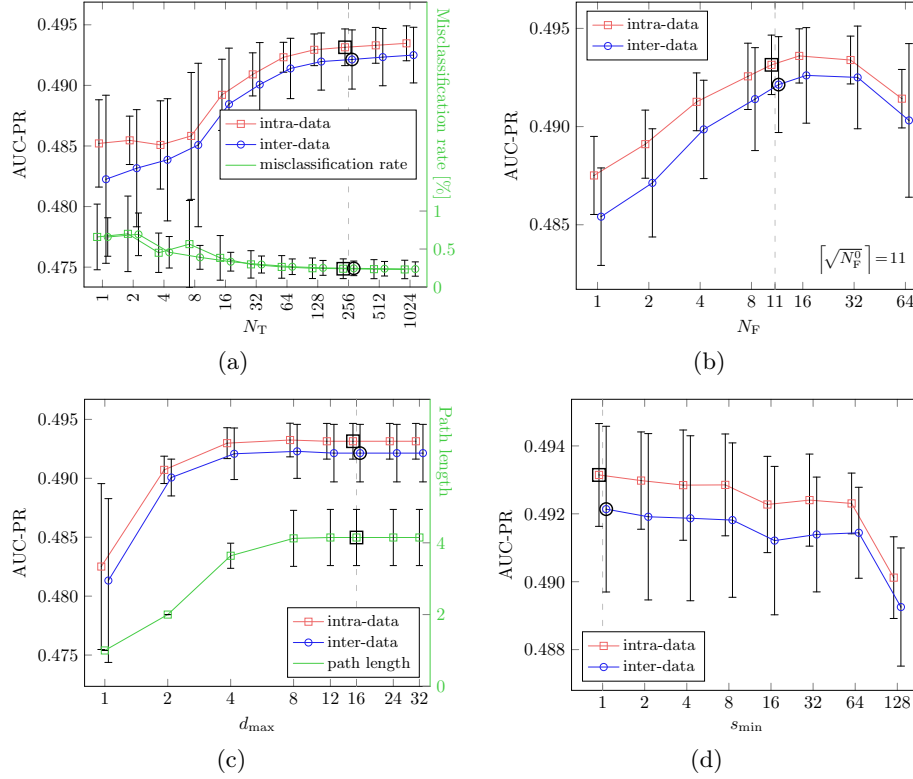


Figure 13: Sensitivity analysis w.r.t. different RF parameters using SFT features: (a) Number of RF trees N_T . (b) Number of features per split N_F . (c) Maximum tree depth d_{\max} . (d) Minimum number of samples stored at leaf node s_{\min} . The partial area under the precision-recall curve (AUC-PR) is computed on the recall interval $[0.5, 1]$. The average numbers are plotted for inter- and intra-data validation (see text) along with the standard deviation indicated by the error bars. Samples at the default parameter values (dashed line) are highlighted with bold black marks. Note that the curves are slightly shifted for better visibility. The RF-SFT model performs robustly for a wide range of parameters.

small average path length. The second break criterion during RF training is the *minimum number of samples* (s_{\min}) stored at each leaf node (see Algorithm 1). The parameter study in Figure 13d shows that the segmentation performance is very stable for a wide range of s_{\min} . Due to the constraint in the linear node model, the split functions can still be robustly estimated during training even for a small number of training exemplars. Similar to the maximum tree depth, disabling the leaf sample restriction during training to obtain fully grown trees ($s_{\min} = 1$) in fact yields the best results. Only for extremely large values ($s_{\min} = 128$), the performance drops significantly as a result of aggressive pruning. Note that the choice of the default parameter values $d_{\max} = 16$ and $s_{\min} = 1$ practically results in fully grown trees without pruning in our experiments.

Regularization of Split Weights. Finally, we investigate the influence of the choice of the most important parameters for the regularized split weight optimization in more detail, namely the *regularization parameter* (λ) and the *elastic net penalty parameter* (α) (see Equations (15) and (16)). The former determines the impact of the regularization term while the latter selects the type of regularization as a compromise between ridge regression ($\alpha = 0$) and the lasso penalty ($\alpha = 1$). Besides the segmentation performance, we also consider the average forest path length as defined in Equation (29) as well as the sparsity of split weights to gain further insight into the effect of the regularization term. For split weights \mathbf{w} , we quantify the sparsity, i.e., number of negligible components, as

$$\text{Sp}(\mathbf{w}) = \frac{1}{|\mathbf{w}|} \sum_{k=1}^{|\mathbf{w}|} \mathbb{1}(|w_k| < \theta_{\text{Sp}} \cdot \|\mathbf{w}\|_{\infty}) \quad , \quad (30)$$

where $\mathbb{1}$ denotes the indicator function, $\|\cdot\|_{\infty}$ the maximum norm, and $\theta_{\text{Sp}} < 1$ a constant threshold.

In a first study, we sweep the regularization parameter λ for constant $\alpha = 0.5$ (default value). Following the concept of pathwise coordinate descent (Friedman et al., 2010), we choose λ relative to the smallest value λ_{\max} for which the covariance updates are nonzero (see Friedman et al. (2010, Section 2.5) for details). In general, the overall segmentation performance slightly drops with increasing (relative) λ , particularly for $\lambda/\lambda_{\max} > 0.5$ (see Figure 14a). As the regularization term gains in importance, the sparsity of the split weights increases significantly while the decision trees become only slightly deeper (see Figure 14b). For $\lambda = 0$, i.e., no regularization at all, the optimization of the split weights seems to perform robust in our case without (obvious) numerical instabilities. The regularization parameter λ is chosen dynamically and varies for each split node as λ_{\max} depends on the training data that is used to estimate the optimal split. A global choice of λ has several drawbacks as indicated by Figures 14c and d. Choosing λ too large can easily make the split estimation instable. As a result, the training process breaks at an early stage leaving incompletely split training data at the leaf nodes and hence producing flatter trees. For instance, virtually all decision trees consist of a single split and two leaf nodes for $\lambda = 1$, which significantly degrades the segmentation performance. Furthermore, the relative choice of λ proves better suited to choose the desired level of sparsity without severely compromising on the segmentation performance as indicated in Figure 14e.

Fixing the value of λ and varying the penalty parameter α , we note that for $\alpha > 0$, the aforementioned effect of increasing split weight sparsity at the expense of a (slight) loss of segmentation performance and deeper trees, is much smaller regardless of the choice of λ (see Figures 14f and g). For $\alpha = 0$, i.e., ridge regression, the sparsity drops significantly while the classification performance remains stable. In our case, ridge regression seems to be able to robustly estimate the (dense) split direction without (obvious) numerical instabilities. Increasing the value of α attaches greater significance to the lasso penalty and hence enforces sparsity in the split weights. It is remarkable that more than half of the split

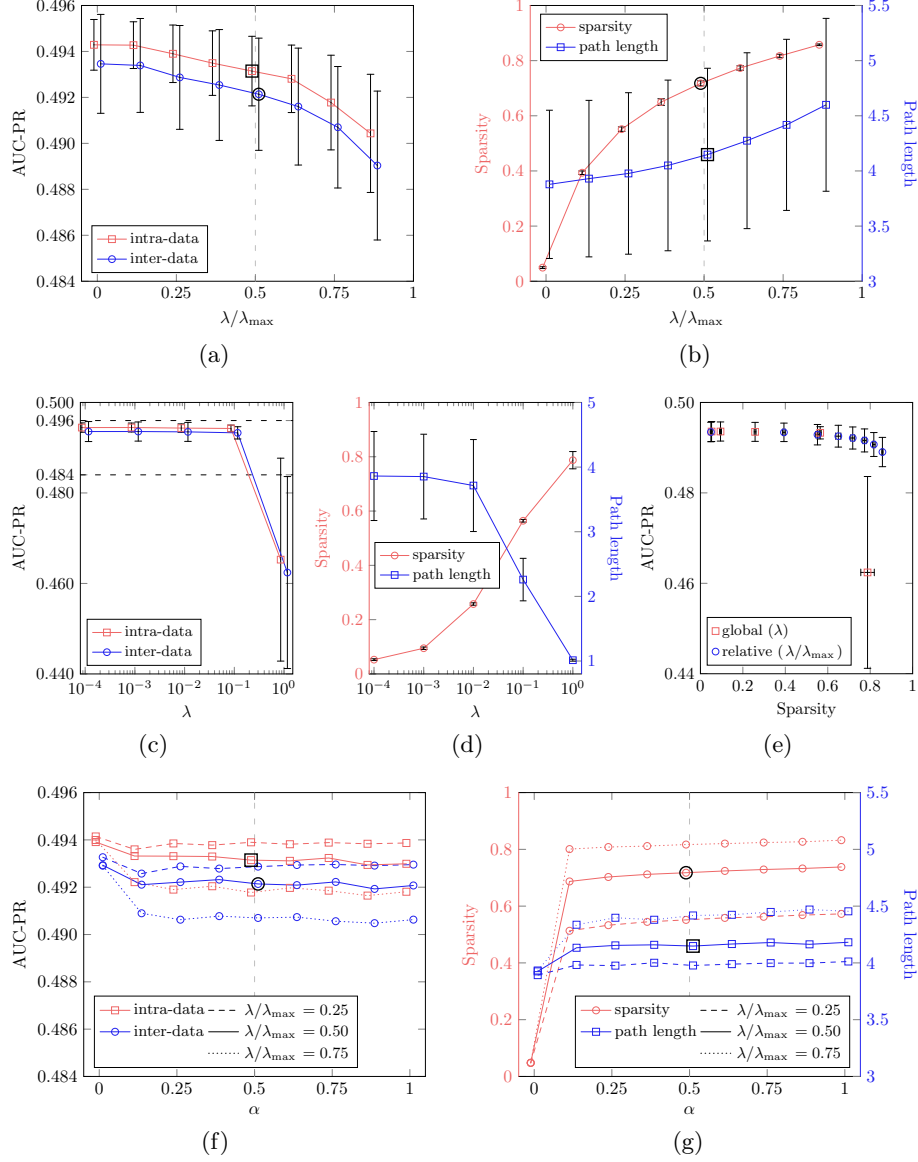


Figure 14: Parameter study of classification forests w.r.t. oblique split parameters (see Equations (15) and (16)). (a,b) Regularization parameter λ relative to λ_{\max} as defined by pathwise coordinate descent (see text). (c,d) Regularization parameter λ . (e) Comparison of global and relative choice of λ . (f,g) Elastic net penalty parameter α for different values of λ/λ_{\max} . We report the normalized average sparsity (g) for an upper threshold of $\theta_{\text{Sp}} = 0.01$ (see text). The lower and upper bounds of the ordinate in (a) and (f) are marked by dashed lines in (c) for better comparability. Also consider the caption of Figure 13 for additional comments. The oblique split parameters allow us to control the sparsity of the split weights at the expense of slightly deeper trees and decreasing segmentation accuracy.

weights are negligible for small values of α without significant impairment of the segmentation performance. This makes the oblique RF more efficient and more similar to univariate splits as the evaluation of the split functions (see Equation (12)) can be accelerated significantly. A proper RF implementation with on-demand feature computation could speed up the RF prediction time even further by extracting the required features only, i.e., feature channels corresponding to nonzero split weights (not used here). At the same time, the average tree depth hardly changes when stepping from ridge regression (no sparsity) to the lasso penalty (maximum sparsity), which leaves the total number of splits to be evaluated during prediction virtually unchanged.

In summary, we find that the learning-based segmentation approaches yield better segmentation results compared to the state-of-the-art baseline approaches, with SFT features being clearly favorable over (r)OSF. Additionally, we find that the oblique split model proves robust and superior to univariate splits. The employed constraint allows for robust estimation of the oblique split direction while maintaining sparsity in the split weights, which can significantly accelerate the evaluation of the split function during prediction.

4.4. Centerline Extraction

In this section, we present and discuss the experimental results for centerline extraction based on the extended RF framework. Considering the superior segmentation performance of the SFT feature model compared to (r)OSF, the same (SFT) features are employed for this task as a second use case.

Qualitative Results and Limitations. The aggregated centerline voting maps for the synthetic dataset are shown in Figure 15 for different configurations. For a low maximum derivative order of the SFT features ($M = 1$), the votes tend to scatter, particularly for large vessels. Including higher order derivatives, the votes become more and more localized and accumulate towards the center of the vessel lumen.

Figure 16 shows qualitative results for the real datasets comparing the different extraction approaches. It can be noted that the OOF approach tends to have some difficulties particularly in regions where the tubular shape model does not hold, e.g., close to bifurcations (see Figures 16a and b) or sudden changes of the vessel radius as in Figures 16b, e, and g. There, the RF- and distance-based extraction perform more robustly. The DOHT centerline shows jagged edges as a result of the discrete skeletonization procedure (see Figures 16c and f). Spline smoothing could be applied to partly overcome this problem (de Boor, 2001). However, it remains unclear how to choose the smoothness parameter and how smoothing eventually affects the accuracy, particularly at tortuous vessel segments. The RF voting scheme allows for robust and highly accurate centerline extraction, in general. Only for very few cases, the RF centerlines locally show small deviations from the ground truth particularly at large segments of set 4 due to missing scales in the training data (see Figure 16f). Visually, the different centerlines can hardly be distinguished from each other in many cases. Therefore, we will present revealing results from a rigorous quantitative analysis next.

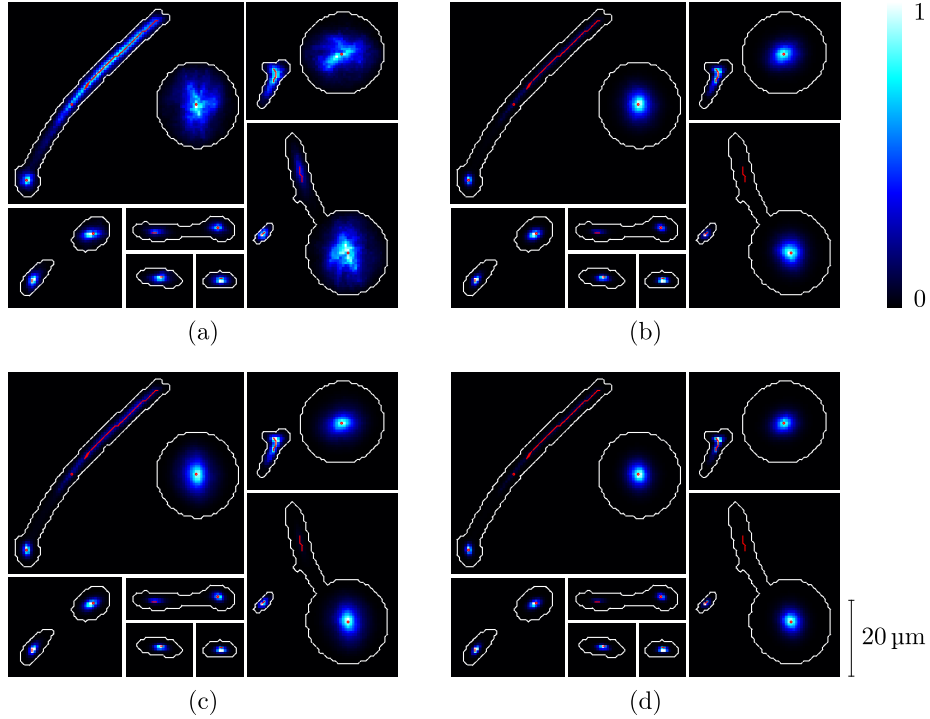


Figure 15: Normalized centerline votes along axial slices of the synthetic test data using SFT features up to order $M = 1$ to $M = 4$ (a,d). The ground truth vessel boundaries and centerlines are superimposed in white and red, respectively. SFT features using first-order derivatives only result in highly scattered centerline votes, particularly for large vessels. Considering higher order derivatives, the accumulated Hough votes become more and more localized and peak at the center of the vessel lumen. [Best viewed in color]

Quantitative Comparison. A quantitative comparison of the different centerline extraction approaches is provided for the synthetic data in Figure 17. The distance-based fast marching scheme (DIST) achieves the best performance on the synthetic dataset. As the distance transform (distance to the background) is computed from the GT segmentation and the vessel segments are modeled as piecewise linear tubular structures, the distance information provides a very strong centerline prior and is hence well suited to drive the wave propagation. However, the skeletonization based on distance-ordered homotopic thinning (DOHT), which utilizes the same distance information computed from the GT labels, performs worse. The oblique Hough forest framework (RF) achieves slightly better accuracy than the OOF-based extraction and performs very robustly as indicated by the small standard deviation. The OOF accuracy particularly decreases for larger segments (see relative error, Figure 17b). We also note that the Hough forest performance shows a similar profile as in the segmentation experiments for varying maximum derivative order of the SFT features (see Figures 17c–d). Considering first-order derivatives only ($M = 1$)

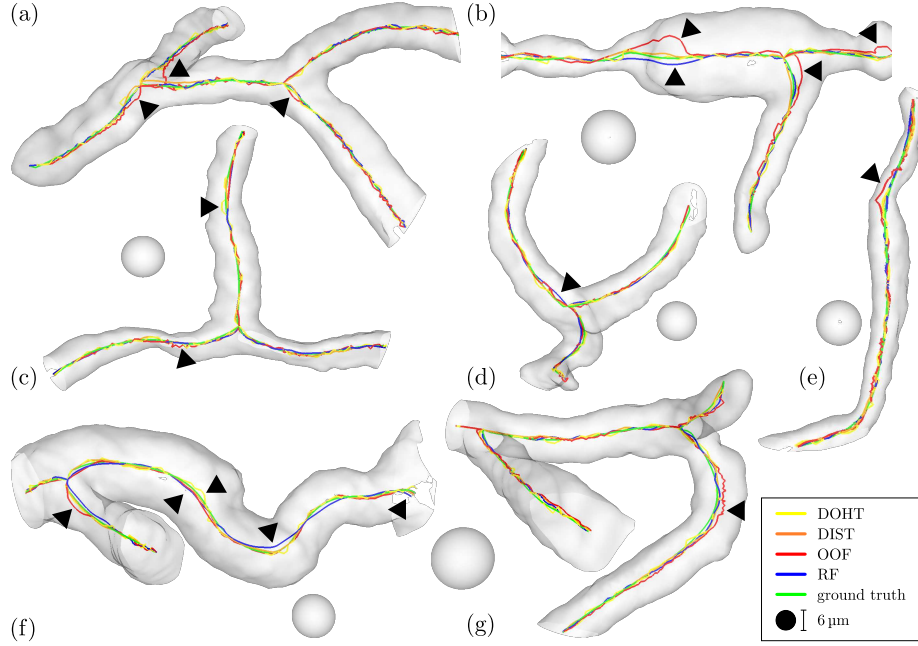


Figure 16: Qualitative comparison of different approaches for centerline extraction on real data (set 1 (a–c), set 2 (d), set 3 (e), set 4 (f–g)). The extracted centerlines are rendered in different colors along with the (opaque) 3-D mesh of the vasculature. Particularly large deviations from the ground truth centerline are marked by arrowheads. The gray spheres have been manually added to indicate the spatial dimensions (sphere diameter $6\mu\text{m}$). [Best viewed in color]

results in a poor performance particularly at large vessels as a result of the rather scattered voting map (see Figure 15a). Incorporating higher order derivatives significantly improves the average centerline accuracy ($M \geq 2$). In fact, similar observations can be made for the experiments on the real datasets.

The centerline accuracy of the leave-one-out cross-validation experiments on the real datasets is summarized in Table 2. In contrast to the synthetic datasets, RF regression performs consistently better than the baseline approaches for all four datasets. The DOHT error reflects the manual adjustments of the expert to define the GT centerline based on the DOHT skeleton. The OOF and distance-based extraction approach work comparably well. The latter shows slightly better absolute errors but tends to be inaccurate for small segments as indicated by the increased relative figures (see also Figure 18). Moreover, it is prone to artifacts of the binary segmentation map which can result in large deviations and leads to the by far largest standard deviation of the compared methods. The RF scheme, however, overcomes these difficulties by vote aggregation and performs much more stable. Only the absolute error and standard deviation for set 4 are slightly increased compared to set 1–3, yet still superior to the other approaches. As mentioned before, this results from missing scales in the training examples randomly sampled from set 1–3 with a smaller radius range listed in

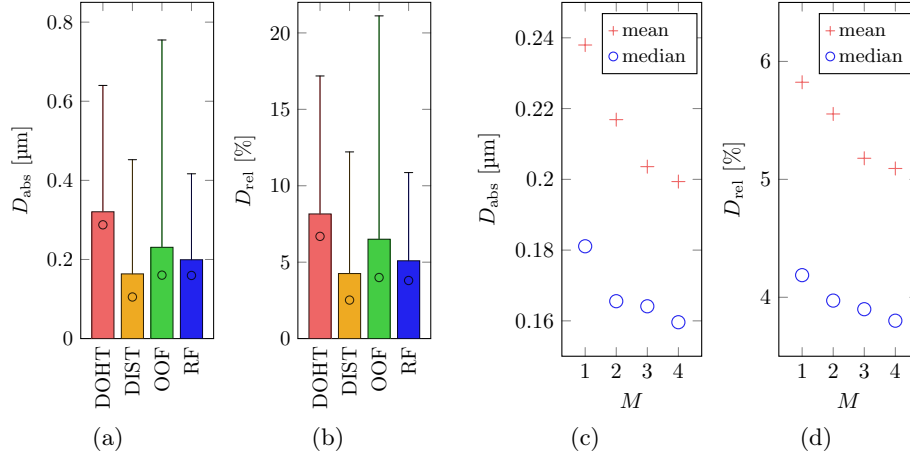


Figure 17: Spatial accuracy of the extracted centerlines for the synthetic data w.r.t. the absolute (a,c) and relative (b,d) error metric D_{abs} and D_{rel} as defined in Equation (28). (a–b) Comparison of the RF framework using Hough regression to the baseline approaches. (c–d) Hough regression for different maximum derivative order M . The median and the standard deviation are indicated by the circular mark and the (unidirectional) error bars, respectively.

Test set (vessel radius)	DOHT	DIST	OOF	RF-SFT
# 1 (0.691–3.17) μm	(0.266 ± 0.153) μm (12.8 ± 7.77) %	(0.366 ± 0.973) μm (19.0 ± 57.4) %	(0.310 ± 0.225) μm (15.1 ± 11.6) %	(0.205 ± 0.144) μm (9.83 ± 7.41) %
# 2 (1.10–4.65) μm	(0.263 ± 0.149) μm (11.4 ± 7.16) %	(0.278 ± 0.574) μm (12.2 ± 28.8) %	(0.315 ± 0.273) μm (13.8 ± 11.6) %	(0.222 ± 0.190) μm (9.31 ± 7.59) %
# 3 (1.21–5.00) μm	(0.270 ± 0.153) μm (12.5 ± 7.87) %	(0.277 ± 0.626) μm (13.8 ± 40.2) %	(0.286 ± 0.184) μm (13.4 ± 9.26) %	(0.219 ± 0.167) μm (9.98 ± 7.60) %
# 4 (0.784–6.70) μm	(0.271 ± 0.154) μm (10.3 ± 6.48) %	(0.316 ± 0.948) μm (11.9 ± 36.0) %	(0.327 ± 0.388) μm (12.0 ± 11.2) %	(0.226 ± 0.209) μm (8.34 ± 6.85) %

Table 2: Quantitative comparison of the centerline accuracy for the four high-resolution datasets w.r.t. the absolute and relative error metric (mean ± standard deviation). The figures highlighted in bold face mark the best result w.r.t. the corresponding metric. The oblique Hough forest framework (RF) performs consistently better than the baseline approaches on all four datasets.

the first column of Table 2.

A more detailed quantitative analysis of the centerline accuracy w.r.t. the structure size of the real datasets is provided in Figure 18. The error histograms confirm the superior performance of Hough forests over the OOF and distance-based extraction approach. For very small vessel segments ($d < 4\mu\text{m}$), the small number of aggregated centerline votes results in a slight loss of accuracy for RF-SFT. In the diameter range of the most frequent vessel segments, i.e. capillaries ($4\mu\text{m} \leq d \leq 6\mu\text{m}$), however, RF-SFT clearly outperforms the other approaches including DOHT with the most stable error profile across all scales

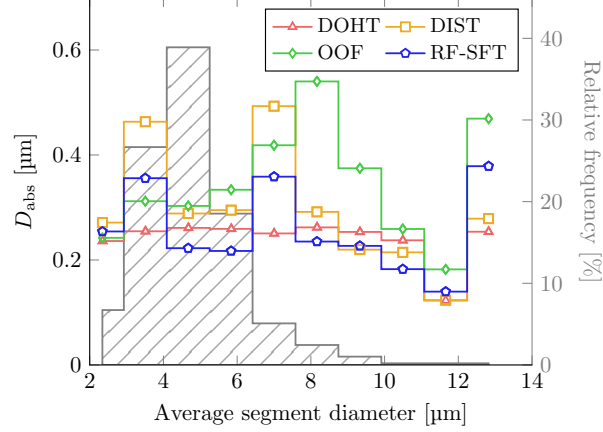


Figure 18: Centerline accuracy as a function of the structure size. Stair plots indicate the average centerline error D_{abs} for the real datasets. The normalized histogram of the vessel segment diameters is superimposed in gray (shaded area). [Best viewed in color]

as the DOHT skeletonization is based on the GT segmentation.

4.5. Sensitivity Analysis of the Hough Regression Forest

We further investigated the sensitivity of the centerline accuracy w.r.t. to the *number of regression trees* (N_T) and the *maximum tree depth* (d_{max}) for the real datasets (see Figure 19). The (relative) error decreases with an increasing number of trees as shown in Figure 19a. For $N_T \geq 32$ the improvements become less drastic and the error rate stabilizes for all four datasets. Similarly, the maximum tree depth d_{max} critically influences the error rate (see Figure 19b). Pruning the fully grown trees reduces the average tree depth, i.e., the number of splits to be evaluated in order to push a sample from the root to a leaf node. At the same time, both the average number of samples per leaf node and the leaf uncertainty as defined in Equation (20) increase (see Figure 19c), which impairs the centerline accuracy. For $d_{\text{max}} \geq 22$, the regression trees become fully grown and the performance converges. Note that the average number of leaf samples remains greater than one as the training exemplars are sampled with replacement for each tree. Duplicate training examples inevitably end up in the same leaf node regardless of the maximum tree depth. The leaf uncertainty, i.e., the variance of the offsets stored at each leaf node, drops to zero in fully grown trees. In this case, all leaf node votes of the different regression trees are weighted equally during vote accumulation (see Equation (21)). The leaf weight defined in Equation (23) becomes uniform $p(l) = 1/N_T$ and hence independent of the shape parameter η , in particular.

In summary, the proposed extraction approach based on Hough regression forests is robust and able to extract highly accurate centerlines for a wide parameter range. Similar to the selection of proper scale parameters for any multi-scale approach, e.g., optimally oriented flux, Hessian-based filters, SFT

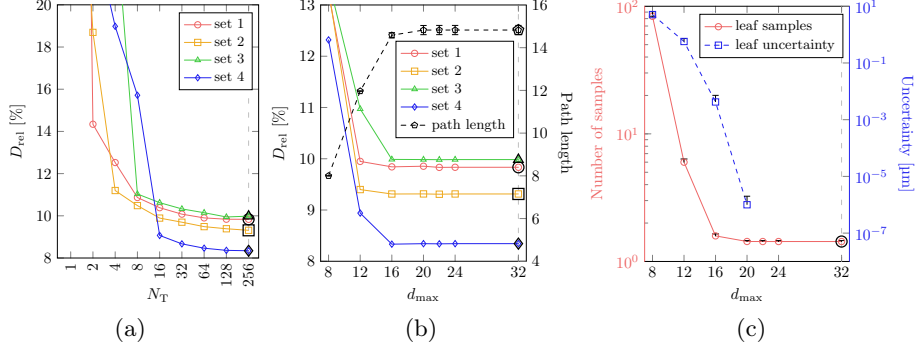


Figure 19: Sensitivity analysis of the extended regression framework w.r.t. different parameters: (a) Number of RF trees N_T . (b-c) Maximum tree depth d_{max} . The average path length (b), the leaf sample uncertainty (see Equation (20)), and the number of samples per leaf node (c) are averaged over the four datasets where the standard deviation is indicated by the (positive) error bars. Note that for $d_{max} \geq 22$ the leaf uncertainty drops to zero. The corresponding samples are hence excluded from the semi logarithmic plot in (c). Samples at the default parameter values (dashed line) are highlighted with bold black marks.

features, care has to be taken to provide proper training data comprising the range of vessel calibers to be processed. As the same (steerable) features are used for both classification and regression, the additional computational cost for centerline extraction is marginal. Ultimately, we can envision a unified framework using decision trees optimized with a multi objective in order to concurrently solve the classification and regression task as in Gall et al. (2011). This could potentially allow for even more efficiently structured trees exploiting the discriminative structure of the common feature space.

5. Summary and Conclusions

We have proposed a machine learning-based framework for accurate and efficient 3-D vessel segmentation and centerline extraction based on oblique random forests (RFs) and Hough voting. For the classification task, we have compared two kinds of features computed from orthogonal subspace filtering and steerable filters, respectively. The latter allow to efficiently decompose the image into a multi-scale rotational basis using steerable filter theory (Jacob and Unser, 2004; González et al., 2009b). Our experiments on 3-D high-resolution imaging data of the rat visual cortex reveal the benefit of steerable over eigenspace filters. Likewise, the introduced oblique split model based on linear regression with an elastic net regularization proves to be superior to univariate orthogonal splits. Furthermore, the constrained optimization of the oblique splits provides an effective parametrization for the sparsity of the split coefficients, which allows us to decide on the level of sparsity, i.e., more efficient split evaluation during prediction at the expense of a slight loss of accuracy. A sensitivity analysis w.r.t. different RF parameters revealed that the segmentation performance remains

stable for a wide parameter range. Moreover, the machine learning-based approach outperforms different state-of-the-art segmentation approaches leveraging the Hessian eigenstructure and optimally oriented flux, respectively. The RF classifiers show excellent classification performance on the 3-D datasets even for incomplete and noisy training labels as obtained by Otsu’s method in our experiments, which renders tedious manual labeling superfluous. The proposed segmentation framework hence allows to fully automatically learn RF models for 3-D vessel segmentation on new datasets. For instance, in our recent study (Rempfler et al., 2014) we found the proposed approach to generalize well for different types of 3-D angiographic whole brain datasets acquired by micro magnetic resonance angiography (μ MRA) or micro computed tomography (μ CT) at different spatial resolution.

We have further demonstrated how to leverage the same steerable features as used for vessel segmentation in an extended regression framework to learn the location of the vessel centerline. To this end, we train a Hough regression forest to infer probabilistic votes about the supposable vessel center at each voxel. The accumulated centerline votes then drive a fast marching scheme in order to extract the most likely centerlines as the minimal path with lowest energy. Our validation experiments on synthetic and real datasets have shown the advantage of the proposed approach over state-of-the-art techniques for centerline extraction w.r.t. centerline accuracy and robustness, particularly in regions where the tubular shape model does not hold, e.g., in the vicinity of bifurcations.

For future work, it would be interesting to further investigate if the combination of different types of features is able to leverage their respective strengths and boost the performance as in Rigamonti and Lepetit (2012). Furthermore, besides centerline extraction, the regression framework could easily be adjusted to learn and predict even more complex information, e.g., vessel caliber, in a general and computationally cheap fashion from local features. The classification framework could be extended along the same lines. Based on the findings of Zhou et al. (2007), combining SFT features in the local neighborhood of each voxel might, for instance, allow for efficient and accurate 3-D bifurcation detection. Another line of research could examine to what extent the concept of auto-context (Tu and Bai, 2010) is able to further boost the segmentation and regression performance. The discriminative vessel and centerline confidence maps provide rich context information, in addition to the original image patches, that can readily be used to train a new classifier. Likewise, similar to Gall et al. (2011), a unified regression forest could be trained to allow for joint classification and regression, i.e., prediction of the class label (foreground vs. background) and inference of the centerline offset in our application. This requires a more complex training strategy, e.g., multi-objective split optimization in order to achieve both purity of class labels and minimal regression uncertainty at the leaf nodes. Additional comprehensive experiments have to be performed to investigate the potential benefit of a single unified classifier over the two-stage RF framework w.r.t. efficiency and accuracy, in particular.

Acknowledgments

This work has been performed within the frame of the Swiss National Center of Competence in Research on Computer Aided and Image Guided Medical Interventions (NCCR Co-Me) supported by the Swiss National Science Foundation. Besides, this research has been supported by the Technische Universität München, Institute for Advanced Study, funded by the German Excellence Initiative and the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no 291763 – the Marie Curie COFUND program of the the European Union.

	Parameter	Value	Description	Text
(r)OSF	P	19 px	Patch size	Eq. 1
	N_P	1.8×10^4	Number of patches	Eq. 1
	$d = N_F^0$	102	PCA dimension (=total number of features)	Eq. 1
SFT	M	4	Maximum derivative order	Eq. 3
	d_M	34	Number of Gaussian derivatives up to order M	Eq. 4
	S	3	Number of scales	Eq. 5
	σ	$(0.70, 1.7, 4.2) [\mu\text{m}]^{\S}$ $(2.0, 4.4, 10) [\mu\text{m}]^{\P}$	SFT scales	Eq. 5
	N_F^0	102 ($= d_M \cdot S$)	Total number of features	Alg. 1
RF	$ \mathcal{T} $	$2 \cdot 4000^{\dagger}$ $32,000^{\ddagger}$ oblique † orthogonal ‡	Number of training samples	Alg. 1
	N_T	256^{\dagger}	Number of RF decision trees	Alg. 1
	d_{\max}	16^{\dagger}	Maximum tree depth	Alg. 1
	d_{\max}	32^{\ddagger}		
	s_{\min}	1	Minimum number of samples per leaf node	Alg. 1
	N_F	1^+ $11^{*\dagger}$ $35^{*\ddagger}$	Number of features per split $(= \lceil \sqrt{N_F^0} \rceil)$ $(= \lceil \frac{N_F^0}{3} \rceil)$	Eq. 12
	λ	$\frac{1}{2} \lambda_{\max}^{\dagger}$ 0.01^{\ddagger}	Regularization parameter	Eq. 15 Eq. 19
	α	0.5	Elastic net penalty parameter	Eq. 16
	η	$10 \frac{\ln(2)}{\Delta}^{\ddagger} \approx 9.9 \mu\text{m}^{-1\S}$ $\approx 6.9 \mu\text{m}^{-1\P}$	Leaf prior parameter relative to isotropic voxel spacing Δ	Eq. 23
	\S Real data (SRXTM) $+$ Orthogonal split \dagger RF classification \P Synthetic data $*$ Oblique split \ddagger Hough forest regression			

Table A.3: Default parameters for experiments

Appendix A. Default Parameters

Default parameters are listed in Table A.3.

Appendix B. Steerability of Gaussian Derivatives

The steerability of Gaussian derivatives has been derived for the 2-D case in Jacob and Unser (2004) and can readily be extended to 3-D (Freeman and Adelson, 1991; González et al., 2009a). Steerability refers to the property that the convolution of an image with a rotated version of the steerable filter template (SFT) can be expressed by a linear combination of the filter response

of the image with the SFT without rotation:

$$I * G_{m,a,b}^\sigma(R\mathbf{x}) = \sum_{i=0}^m \sum_{j=0}^i \omega_{m,a,b}^{i,j} \underbrace{(I * G_{m,i,j}^\sigma)(\mathbf{x})}_{\mathbf{f}_{m,i,j}^\sigma(I,\mathbf{x})} \quad , \quad (\text{B.1})$$

where $R \in SO(3)$ denotes a 3-D rotation matrix:

$$R_{\theta,\phi} = \begin{pmatrix} \cos \theta \cos \phi & \cos \theta \sin \phi & \sin \theta \\ -\sin \phi & \cos \phi & 0 \\ -\sin \theta \cos \phi & -\sin \theta \sin \phi & \cos \theta \end{pmatrix} \quad . \quad (\text{B.2})$$

The uniquely defined coefficients $\omega_{m,a,b}^{i,j}$ can be computed in closed form as:

$$\begin{aligned} w_{m,a,b}^{i,j} = \sum_{\mathbf{s} \in \mathcal{P}_{m,a,b}^{i,j}} & (-1)^{a-v_1-w_2} \binom{m-a}{u_1} \binom{a-b}{v_1} \binom{b}{w_1} \binom{u_1}{u_2} \binom{w_1}{w_2} \\ & (\cos \theta)^{m-a-u_2+w_2} (\cos \phi)^{m-a+b-u_1+v_1-w_1} \\ & (\sin \theta)^{b+u_2-w_2} (\sin \phi)^{a-b+u_1-v_1+w_1-u_2-w_2} \quad , \quad (\text{B.3}) \end{aligned}$$

where

$$\begin{aligned} \mathcal{P}_{m,a,b}^{i,j} = \{ & (u_1, v_1, w_1, u_2, w_2)^\top \in \mathbb{N}_0^5 \mid u_1 \leq m-a, v_1 \leq a-b, \\ & w_1 \leq b, u_2 \leq u_1, w_2 \leq w_1, \\ & u_1 + v_1 + w_1 = i, u_2 + w_2 = j \} \quad . \quad (\text{B.4}) \end{aligned}$$

Proof. Following Jacob and Unser (2004), the Fourier transformation (FT) of the rotated filter templates as defined in Equations (B.1) and (B.2) can be computed by a rotation in the Fourier domain:

$$\begin{aligned} \mathcal{F}(G_{m,a,b}^\sigma(R_{\theta,\phi}\mathbf{x})) = & (i\omega_x \cos \theta \cos \phi + i\omega_y \cos \theta \sin \phi + i\omega_z \sin \theta)^{m-a} \\ & (-i\omega_x \sin \phi + i\omega_y \cos \phi + 0)^{a-b} \quad (\text{B.5}) \\ & (-i\omega_x \sin \theta \cos \phi + i\omega_y \sin \theta \sin \phi + i\omega_z \cos \theta)^b \cdot \hat{G}^\sigma(\omega) \quad , \end{aligned}$$

where $\hat{G}^\sigma(\mathbf{x}) = \mathcal{F}(G^\sigma(\mathbf{x}))$ denotes the transfer function of the Gaussian kernel. Repeatedly applying the binomial equation, this can be rewritten as:

$$\begin{aligned} \mathcal{F}(G_{m,a,b}^\sigma(R_{\theta,\phi}\mathbf{x})) = & \sum_{u_1=0}^{m-a} \sum_{v_1=0}^{a-b} \sum_{w_1=0}^b \sum_{u_2=0}^{u_1} \sum_{w_2=0}^{w_1} \psi_{m,a,b}^{(u_1,v_1,w_1,u_2,w_2)^\top} \cdot \hat{G}^\sigma(\omega) \quad (\text{B.6}) \\ & (i\omega_x)^{m-(u_1+v_1+w_1)} (i\omega_y)^{(u_1+v_1+w_1)-(u_2+w_2)} (i\omega_z)^{u_2+w_2} \quad , \end{aligned}$$

where

$$\begin{aligned} \psi_{m,a,b}^{(u_1,v_1,w_1,u_2,w_2)^\top} = & (-1)^{a-v_1-w_2} \binom{m-a}{u_1} \binom{a-b}{v_1} \binom{b}{w_1} \binom{u_1}{u_2} \binom{w_1}{w_2} \\ & (\cos \theta)^{m-a-u_2+w_2} (\cos \phi)^{m-a+b-u_1+v_1-w_1} \\ & (\sin \theta)^{b+u_2-w_2} (\sin \phi)^{a-b+u_1-v_1+w_1-u_2-w_2} \quad . \quad (\text{B.7}) \end{aligned}$$

Applying the inverse FT to Equation (B.6) and convolving with image I finally yields:

$$\begin{aligned}
I * G_{m,a,b}^\sigma(R_{\theta,\phi}\mathbf{x}) &= \sum_{u_1,v_1,w_1,u_2,w_2} \psi_{m,a,b}^{(u_1,v_1,w_1,u_2,w_2)^\top} (I * G_{m,u_1+v_1+w_1,u_2+w_2}^\sigma)(\mathbf{x}) \\
&= \sum_{i=0}^m \sum_{j=0}^i \underbrace{\sum_{\substack{\mathbf{s} \in \mathcal{P}_{m,a,b}^{i,j} \\ w_{m,a,b}^{i,j}}} \psi_{m,a,b}^{\mathbf{s}}}_{w_{m,a,b}^{i,j}} \underbrace{(I * G_{m,i,j}^\sigma)(\mathbf{x})}_{\mathbf{f}_{m,i,j}(I,\mathbf{x})} \quad , \quad (\text{B.8})
\end{aligned}$$

where

$$\begin{aligned}
\mathcal{P}_{m,a,b}^{i,j} &= \{(u_1, v_1, w_1, u_2, w_2)^\top \in \mathbb{N}_0^5 \mid u_1 \leq m-a, v_1 \leq a-b, \\
&\quad w \leq b, u_2 \leq u_1, w_2 \leq w_1, \\
&\quad u_1 + v_1 + w_1 = i, u_2 + w_2 = j\} \quad . \quad (\text{B.9})
\end{aligned}$$

□

Note that care has to be taken when applying FT-based convolution. The bandwidth of the Gaussian derivatives is governed by the standard deviation σ of the Gaussian function. Without proper oversampling, a small σ can result in signal aliasing and undesired magnification after FT-based convolution (Law and Chung, 2009).

Appendix C. Segmentation Results

The binary segmentation of the cerebrovascular networks of the test ROIs of datasets D_{1-4} are compared for the different vessel segmentation approaches in Figures C.20–C.23.

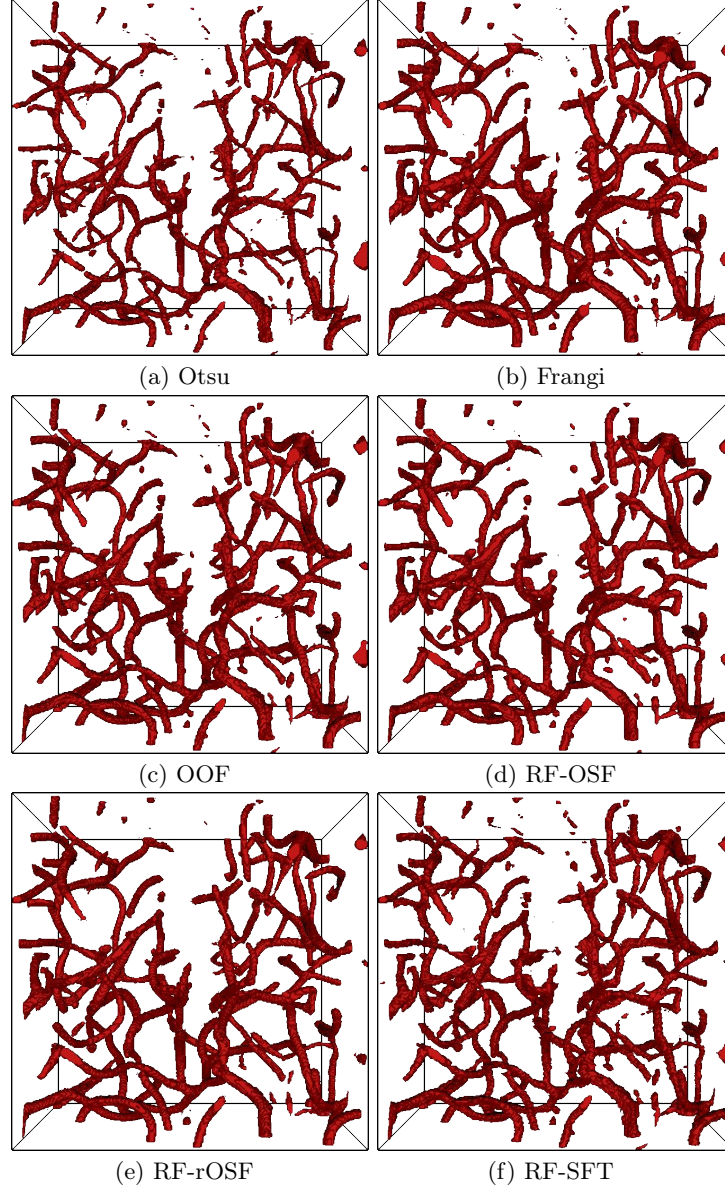


Figure C.20: Segmented cerebrovascular network for the 3-D test ROI of dataset D_1 using different segmentation techniques. (a) Otsu's method (Otsu, 1979). (b) Frangi (Frangi et al., 1998). (c) OOF (Law and Chung, 2008). (d) RF-OSF ($d = 102$). (e) RF-rOSF ($d = 102$). (f) RF-SFT ($M = 4$). The binary segmentation maps are computed at the corresponding F_1 -optimal operating points.

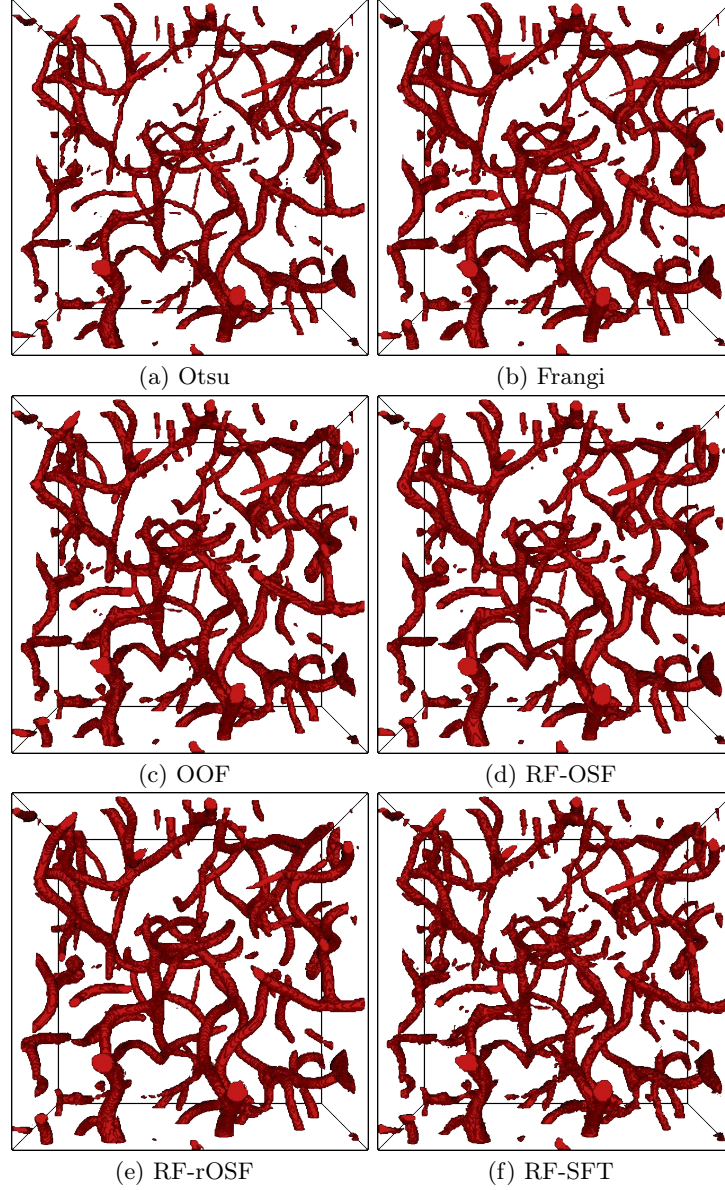


Figure C.21: Segmented cerebrovascular network for the 3-D test ROI of dataset D_2 using different segmentation techniques. (a) Otsu's method (Otsu, 1979). (b) Frangi (Frangi et al., 1998). (c) OOF (Law and Chung, 2008). (d) RF-OSF ($d = 102$). (e) RF-rOSF ($d = 102$). (f) RF-SFT ($M = 4$). The binary segmentation maps are computed at the corresponding F_1 -optimal operating points.

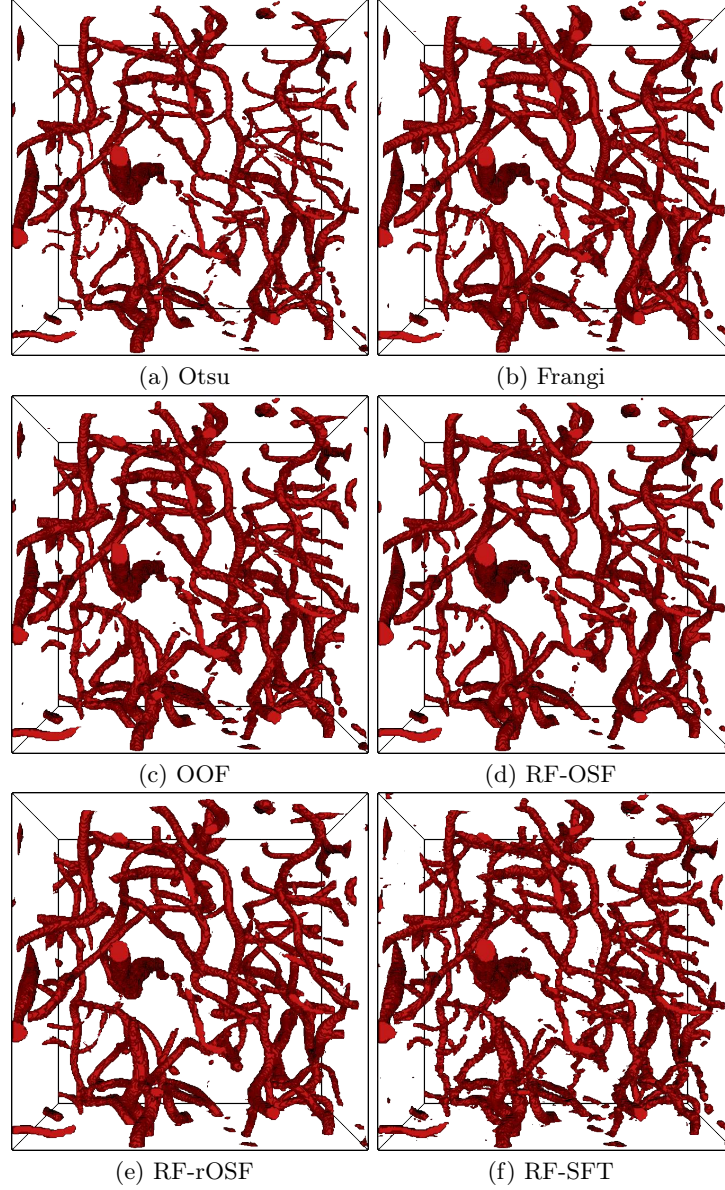


Figure C.22: Segmented cerebrovascular network for the 3-D test ROI of dataset D_3 using different segmentation techniques. (a) Otsu's method (Otsu, 1979). (b) Frangi (Frangi et al., 1998). (c) OOF (Law and Chung, 2008). (d) RF-OSF ($d = 102$). (e) RF-rOSF ($d = 102$). (f) RF-SFT ($M = 4$). The binary segmentation maps are computed at the corresponding F_1 -optimal operating points.

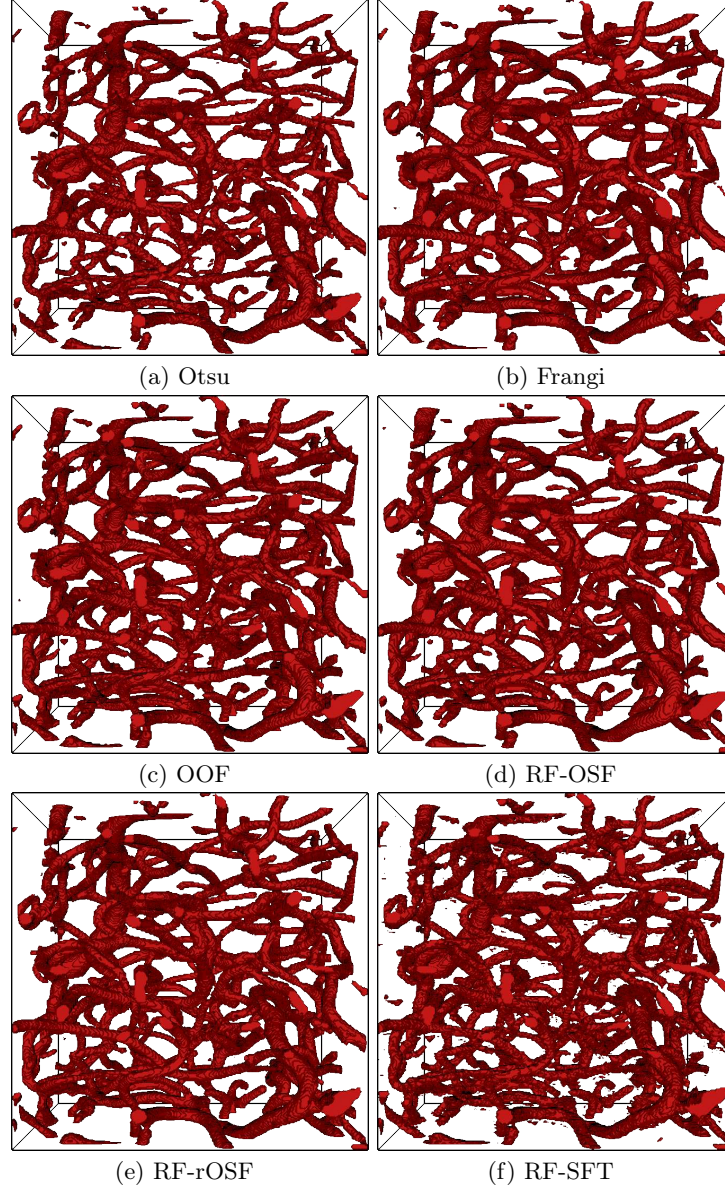


Figure C.23: Segmented cerebrovascular network for the 3-D test ROI of dataset D_4 using different segmentation techniques. (a) Otsu's method (Otsu, 1979). (b) Frangi (Frangi et al., 1998). (c) OOF (Law and Chung, 2008). (d) RF-OSF ($d = 102$). (e) RF-rOSF ($d = 102$). (f) RF-SFT ($M = 4$). The binary segmentation maps are computed at the corresponding F_1 -optimal operating points.

References

- Benmansour, F., Cohen, L.D., 2011. Tubular structure segmentation based on minimal path method and anisotropic enhancement. *International Journal of Computer Vision* 92, 192–210.
- Benmansour, F., Turetken, E., Fua, P., 2013. Tubular geodesics using oriented flux: An ITK implementation. *Insight Journal*.
- de Boor, C., 2001. A Practical Guide to Splines. Number 27 in Applied Mathematical Sciences, Springer Berlin/Heidelberg, New York. Revised edition.
- Breiman, L., 2001. Random forests. *Machine Learning* 45, 5–32.
- Breitenreicher, D., Sofka, M., Britzen, S., Zhou, S.K., 2013. Hierarchical discriminative framework for detecting tubular structures in 3D images, in: Gee, J.C., Joshi, S., Pohl, K.M., Wells, W.M., Zöllei, L. (Eds.), *Information Processing in Medical Imaging (IPMI 2013)*, Springer Berlin/Heidelberg. pp. 328–339.
- Canny, J., 1983. Finding Edges and Lines in Images. Technical Report. Massachusetts Institute of Technology. Cambridge, MA, USA.
- Chen, Y.W., Lin, C.J., 2006. Combining SVMs with various feature selection strategies, in: Guyon, I., Nikravesh, M., Gunn, S., Zadeh, L. (Eds.), *Feature Extraction*, Springer Berlin/Heidelberg. pp. 315–324.
- Chen, Z., Molloy, S., 2003. Automatic 3D vascular tree construction in CT angiography. *Computerized Medical Imaging and Graphics* 27, 469–479.
- Cohen, L.D., Kimmel, R., 1997. Global minimum for active contour models: A minimal path approach. *International Journal of Computer Vision* 24, 57–78.
- Criminisi, A., Shotton, J., Robertson, D., Konukoglu, E., 2011. Regression forests for efficient anatomy detection and localization in CT studies, in: Menze, B., Langs, G., Tu, Z., Criminisi, A. (Eds.), *Medical Computer Vision. Recognition Techniques and Applications in Medical Imaging*, Springer Berlin/Heidelberg. pp. 106–117.
- Deschamps, T., 2001. Curve and Shape Extraction with Minimal Path and Level-Sets Techniques: Applications to 3D Medical Imaging. Ph.D. thesis. Université Paris-Dauphine.
- Florin, C., Paragios, N., Williams, J., 2006. Globally optimal active contours, sequential Monte Carlo and on-line learning for vessel segmentation, in: Leonardis, A., Bischof, H., Pinz, A. (Eds.), *Computer Vision – ECCV 2006*, Springer Berlin/Heidelberg. pp. 476–489.
- Frangi, A., Niessen, W., Vincken, K., Viergever, M., 1998. Multiscale vessel enhancement filtering, in: Wells, W.M., Colchester, A., Delp, S. (Eds.), *Medical Image Computing and Computer-Assisted Intervention – MICCAI’98*, Springer Berlin/Heidelberg. pp. 130–137.

- Freeman, W.T., Adelson, E.H., 1991. The design and use of steerable filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13, 891–906.
- Friedman, J.H., Hastie, T., Tibshirani, R., 2010. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software* 33, 1–22.
- Gall, J., Lempitsky, V., 2009. Class-specific Hough forests for object detection, in: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2009)*, pp. 1022–1029.
- Gall, J., Yao, A., Razavi, N., Van Gool, L., Lempitsky, V., 2011. Hough forests for object detection, tracking, and action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33, 2188–2202.
- González, G., Aguet, F., Fleuret, F., Unser, M., Fua, P., 2009a. Steerable features for statistical 3D dendrite detection, in: Yang, G.Z., Hawkes, D., Rueckert, D., Noble, A., Taylor, C. (Eds.), *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2009*, Springer Berlin/Heidelberg. pp. 625–632.
- González, G., Fleuret, F., Fua, P., 2009b. Learning rotational features for filament detection, in: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2009)*, pp. 1582–1589.
- Gülsün, M., Tek, H., 2008. Robust vessel tree modeling, in: Metaxas, D., Axel, L., Fichtinger, G., Székely, G. (Eds.), *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2008*, Springer Berlin/Heidelberg. pp. 602–611.
- Hastie, T., Tibshirani, R., Friedman, J.H., 2009. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics, Springer. 2nd edition.
- Hernández-Vela, A., Gatta, C., Escalera, S., Igual, L., Martín-Yuste, V., Radeva, P., 2011. Accurate and robust fully-automatic QCA: Method and numerical validation, in: Fichtinger, G., Martel, A., Peters, T. (Eds.), *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2011*, Springer Berlin/Heidelberg. pp. 496–503.
- Hirsch, S., Reichold, J., Schneider, M., Székely, G., Weber, B., 2012. Topology and hemodynamics of the cortical cerebrovascular system. *Journal of Cerebral Blood Flow and Metabolism* 32, 952–967.
- Jacob, M., Unser, M., 2004. Design of steerable filters for feature detection using canny-like criteria. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26, 1007–1019.
- Johnson, H.J., McCormick, M., Ibáñez, L., The Insight Software Consortium, 2013. *The ITK software guide*. Kitware, Inc.. 3rd edition.

- Jolliffe, I.T., 2002. *Principal Component Analysis*. Springer Berlin/Heidelberg. 2nd edition.
- Kirbas, C., Quek, F., 2004. A review of vessel extraction techniques and algorithms. *ACM Computing Surveys* 36, 81–121.
- Krissian, K., Wu, X., Luboz, V., 2006. Smooth vasculature reconstruction with circular and elliptic cross sections. *Studies in Health Technology and Informatics* 119, 273–278.
- Law, M.W.K., Chung, A.C.S., 2008. Three dimensional curvilinear structure detection using optimally oriented flux, in: Forsyth, D., Torr, P., Zisserman, A. (Eds.), *Computer Vision – ECCV 2008*, Springer Berlin/Heidelberg. pp. 368–382.
- Law, M.W.K., Chung, A.C.S., 2009. Efficient implementation for spherical flux computation and its application to vascular segmentation. *IEEE Transactions on Image Processing* 18, 596–612.
- Law, M.W.K., Chung, A.C.S., 2010. An oriented flux symmetry based active contour model for three dimensional vessel segmentation, in: Daniilidis, K., Maragos, P., Paragios, N. (Eds.), *Computer Vision – ECCV 2010*, Springer Berlin/Heidelberg. pp. 720–734.
- Law, M.W.K., Tay, K., Leung, A., Garvin, G.J., Li, S., 2013. Intervertebral disc segmentation in MR images using anisotropic oriented flux. *Medical Image Analysis* 17, 43–61.
- Lesage, D., Angelini, E., Bloch, I., Funka-Lea, G., 2008. Medial-based Bayesian tracking for vascular segmentation: Application to coronary arteries in 3D CT angiography, in: *IEEE International Symposium on Biomedical Imaging: From Nano to Macro (ISBI 2008)*, pp. 268–271.
- Lesage, D., Angelini, E.D., Bloch, I., Funka-Lea, G., 2009. A review of 3D vessel lumen segmentation techniques: Models, features and extraction schemes. *Medical Image Analysis* 13, 819–845.
- Liaw, A., Wiener, M., 2002. Classification and regression by randomforest. *R News* 2, 18–22.
- Lindeberg, T., 1996. Edge detection and ridge detection with automatic scale selection. *International Journal of Computer Vision* 30, 465–470.
- Lo, P., Sporring, J., Ashraf, H., Pedersen, J.J., de Bruijne, M., 2010. Vessel-guided airway tree segmentation: A voxel classification approach. *Medical Image Analysis* 14, 527–538.
- Lorigo, L., Faugeras, O., Grimson, W., Keriven, R., Kikinis, R., Nabavi, A., Westin, C.F., 2001. CURVES: Curve evolution for vessel segmentation. *Medical Image Analysis* 5, 195–206.

- Martínez-Pérez, M.E., Hughes, A.D., Stanton, A.V., Thom, S.A., Bharath, A.A., Parker, K.H., 1999. Retinal blood vessel segmentation by means of scale-space analysis and region growing, in: Taylor, C., Colchester, A. (Eds.), *Medical Image Computing and Computer-Assisted Intervention – MICCAI’99*, Springer Berlin/Heidelberg. pp. 90–97.
- Menze, B.H., Kelm, B.M., Hamprecht, F.A., 2006. From eigenspots to fisherspots – Latent spaces in the nonlinear detection of spot patterns in a highly varying background, in: Decker, R., Lenz, H.J. (Eds.), *Advances in Data Analysis*, Springer Berlin/Heidelberg. pp. 255–262.
- Menze, B.H., Kelm, B.M., Splittho, N., Koethe, U., Hamprecht, F.A., 2011. On oblique random forests, in: *European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD 2011)*, Springer Berlin/Heidelberg. pp. 453–469.
- Moon, T., Stirling, W., 2000. *Mathematical Methods and Algorithms for Signal Processing*. Prentice Hall.
- Otsu, N., 1979. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics* 9, 62–66.
- Palágyi, K., Kuba, A., 1998. A 3D 6-subiteration thinning algorithm for extracting medial lines. *Pattern Recognition Letters* 19, 613–627.
- Perona, P., Malik, J., 1990. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 12, 629–639.
- Pudney, C., 1998. Distance-ordered homotopic thinning: A skeletonization algorithm for 3D digital images. *Computer Vision and Image Understanding* 72, 404–413.
- Reichold, J., Stampanoni, M., Keller, A.L., Buck, A., Jenny, P., Weber, B., 2009. Vascular graph model to simulate the cerebral blood flow in realistic vascular networks. *Journal of Cerebral Blood Flow and Metabolism* 29, 1429–1443.
- Rematas, K., Leibe, B., 2011. Efficient object detection and segmentation with a cascaded Hough forest ISM, in: *IEEE Workshop on Challenges and Opportunities in Robot Perception (ICCV 2011)*, pp. 966–973.
- Rempfler, M., Schneider, M., Ielacqua, G., Xiao, X., Stock, S.R., Klohs, J., Székely, G., Andres, B., Menze, B.H., 2014. Extracting vascular networks under physiological constraints via integer programming, in: Golland, P., Hata, N., Barillot, C., Hornegger, J., Howe, R. (Eds.), *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2014*, Springer Berlin/Heidelberg. pp. 506–513.

- Rigamonti, R., Lepetit, V., 2012. Accurate and efficient linear structure segmentation by leveraging ad hoc features with learned filters, in: Ayache, N., Delingette, H., Golland, P., Mori, K. (Eds.), *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2012*, Springer Berlin/Heidelberg. pp. 189–197.
- Rigamonti, R., Türetken, E., González Serrano, G., Fua, P., Lepetit, V., 2011. Filter learning for linear structure segmentation. Technical Report. Ecole Polytechnique Fédéral de Lausanne (EPFL), Lausanne, Switzerland.
- Risser, L., Plouraboue, F., Descombes, X., 2008. Gap filling of 3-D microvascular networks by tensor voting. *IEEE Transactions on Medical Imaging* 27, 674–687.
- Rouchdy, Y., Cohen, L.D., 2012. A geodesic voting shape prior to constrain the level set evolution for the segmentation of tubular trees, in: Bruckstein, A.M., Haar Romeny, B.M., Bronstein, A.M., Bronstein, M.M. (Eds.), *Scale Space and Variational Methods in Computer Vision (SSVM 2011)*, Springer Berlin/Heidelberg. pp. 362–373.
- Sato, Y., Nakajima, S., Atsumi, H., Koller, T., Gerig, G., Yoshida, S., Kikinis, R., 1997. 3D multi-scale line filter for segmentation and visualization of curvilinear structures in medical images, in: Troccaz, J., Grimson, E., Mösges, R. (Eds.), *Computer Vision, Virtual Reality and Robotics in Medicine and Medical Robotics and Computer-Assisted Surgery (CVRMed-MRCAS’97)*, Springer Berlin/Heidelberg. pp. 213–222.
- Schneider, M., Hirsch, S., Székely, G., Weber, B., Menze, B.H., 2013. Oblique random forests for 3-D vessel detection using steerable filters and orthogonal subspace filtering, in: Menze, B.H., Langs, G., Montillo, A., Tu, Z., Criminisi, A. (Eds.), *MICCAI Workshop on Medical Computer Vision (MCV 2012)*, Springer Berlin/Heidelberg. pp. 142–154.
- Schneider, M., Hirsch, S., Weber, B., Székely, G., Menze, B.H., 2014. TGIF: Topological gap in-fill for correction of vascular connectivity – A generative physiological modeling approach, in: Golland, P., Hata, N., Barillot, C., Hornegger, J., Howe, R. (Eds.), *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2014*, Springer Berlin/Heidelberg. pp. 89–96.
- Schneider, M., Reichold, J., Weber, B., Székely, G., Hirsch, S., 2012. Tissue metabolism driven arterial tree generation. *Medical Image Analysis* 16, 1397–1414. Special Issue on the 2011 Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI 2011).
- Schneider, M., Sundar, H., 2010. Automatic global vessel segmentation and catheter removal using local geometry information and vector field integration, in: *IEEE International Symposium on Biomedical Imaging: From Nano to Macro (ISBI 2010)*, pp. 45–48.

- Schneider, N., 1990. Kantenhervorhebung und Kantenverfolgung in der industriellen Bildverarbeitung. volume 6 of *Fortschritte der Robotik*. Vieweg+Teubner Verlag.
- Sethian, J., 1999. Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science. Cambridge Monograph on Applied and Computational Mathematics, Cambridge University Press. 2nd edition.
- Tu, Z., Bai, X., 2010. Auto-context and its application to high-level vision tasks and 3D brain image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32, 1744–1757.
- Türetken, E., Benmansour, F., Fua, P., 2012. Automated reconstruction of tree structures using path classifiers and mixed integer programming, in: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2012)*, pp. 566–573.
- Xiao, C., Staring, M., Wang, Y., Shamonin, D., Stoel, B., 2013. Multiscale bi-Gaussian filter for adjacent curvilinear structures detection with application to vasculature images. *IEEE Transactions on Image Processing* 22, 174–188.
- Yushkevich, P.A., Piven, J., Hazlett, H.C., Smith, R.G., Ho, S., Gee, J.C., Gerig, G., 2006. User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability. *NeuroImage* 31, 1116–1128. <http://www.itksnap.org>.
- Zhou, J., Chang, S., Metaxas, D., Axel, L., 2007. Vascular structure segmentation and bifurcation detection, in: *IEEE International Symposium on Biomedical Imaging: From Nano to Macro (ISBI 2007)*, pp. 872–875.